

**CHURCH-ROSSER THEOREM UND
NORMALISIERUNG FÜR TERMKALKÜLE MIT
UNENDLICHEN TERMEN UNTER EINSCHLUSS
PERMUTATIVER REDUKTIONEN**

Martin Ruckert

München 1985

CHURCH-ROSSER THEOREM und NORMALISIERUNG für
TERMKALKÜLE mit UNENDLICHEN TERMEN
unter EINSCHLUSS PERMUTATIVER REDUKTIONEN.

Dissertation an der Fakultät für Mathematik
an der Ludwig-Maximilians-Universität München

vorgelegt von
Martin Ruckert
am 1. Mai 1985

1. Berichterstatter: Prof. Dr. H. Schwichtenberg

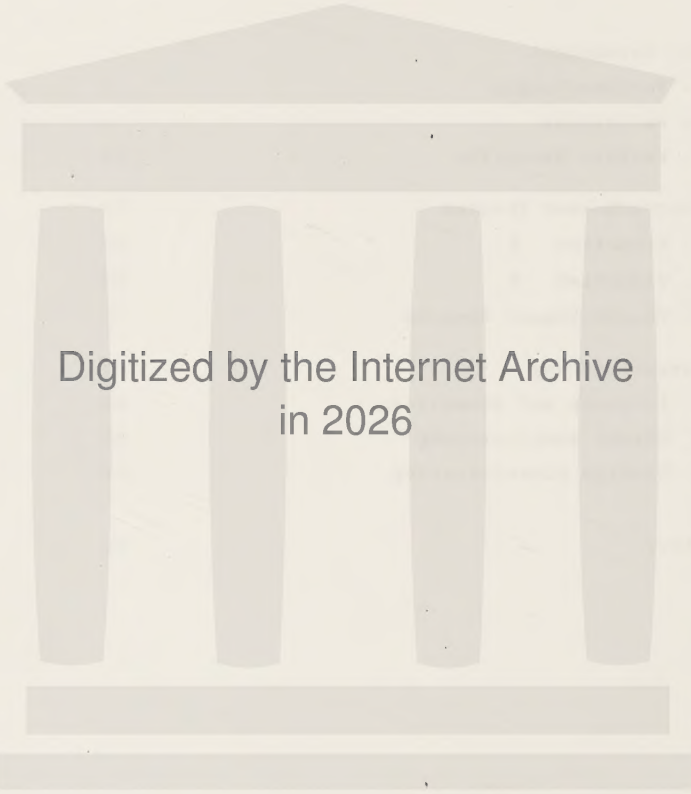
2. Berichterstatter: Prof. Dr. W. Buchholz

Tag der mündlichen Prüfung: 22. Juli 1985

Für Herrn Dr. Herbert Wallbrecher,
ohne den diese Arbeit nicht entstanden wäre.

CHURCH-ROSSER THEOREM und NORMALISIERUNG für
TERMKALKÜLE mit UNENDLICHEN TERMEN
unter EINSCHLUSS PERMUTATIVER REDUKTIONEN.

Inhalt:	Seite
I Der Termkalkül	
1. Vorbemerkungen	1
2. Definition	3
3. Weitere Beispiele	14
II Church-Rosser Theorem	24
1. Reduktion $\rightarrow$	25
2. Reduktion $\Rightarrow$	33
3. Church-Rosser Theorem	35
III Normalisierung	40
1. Existenz der Normalform	40
2. Starke Normalisierung	48
3. Stetige Normalisierung	62
Literatur	85



Digitized by the Internet Archive
in 2026

Zusammenfassung:

Bei der Arbeit mit Termen und Reduktionskalkülen stellen sich stets zwei Probleme: Die Frage nach der Existenz einer Normalform und, gegebenenfalls, der Eindeutigkeit dieser Normalform.

Besonders schwierig werden diese Fragen sobald man unendliche Termbildungsregeln und permutative Reduktionsschritte zuläßt.

Ziel der vorliegenden Dissertation ist es diese beiden Fragen in einem allgemeinen Rahmen zu untersuchen.

So wird ein allgemeiner Termkalkül vorgestellt, der als Basis für die folgenden Untersuchungen dient. Seine Anwendung wird durch Beispiele (λ -Kalkül, Rekursion, natürliche Herleitungen) erläutert. Der Kalkül leistet die Analyse des Reduktionsvorganges durch den Begriff der Einführung, der Beseitigung und der kritischen Beseitigung.

Im zweiten Kapitel wird für den allgemeinen Kalkül ein Church-Rosser Theorem bewiesen.

Das dritte Kapitel gibt zunächst einen Existenzbeweis für Normalformen in diesem Kalkül. Sodann wird für einen Spezialfall des Kalküls die Frage der starken Normalisierung untersucht.

Der letzte Teil enthält schließlich ein Verfahren zur stetigen Normalisierung solcher allgemeinen Termkalküle.

Während beim Existenz- und Eindeutigkeitssatz in der Literatur immerhin Spezialfälle ähnlicher Komplexität behandelt sind, ist ein Verfahren zur stetigen Normalisierung, von vergleichbarer Stärke, bisher nicht bekannt.

1. Der Termkalkül

1. Vorbemerkungen

Hat man das Ziel, ein "allgemeines" Church-Rosser Theorem zu beweisen, so steht man zuerst vor dem Problem, genau festlegen zu müssen, was "allgemein" heißen soll.

Man möchte ja einerseits die vielen Fälle, in denen man ein Church-Rosser Theorem braucht, möglichst alle erfassen, andererseits muß der Kreis so eng gezogen sein, daß wenigstens die Fälle draußen bleiben, für die das Theorem nicht gilt.

Gleichzeitig sollte das Theorem so einfach sein, daß es leicht anwendbar ist. (Die Anwendung auf ein gegebenes System sollte ja nicht schwieriger sein als das Theorem selbst, sondern vergleichsweise einfach.)

Betrachtet man Beweise über Termkalküle, so fällt zuerst auf, daß die Beweisschemata stets die gleichen sind: Induktion und Fallunterscheidung. Auch in der vorliegenden Arbeit ist dies nicht anders. Bei den Fallunterscheidungen, die oft einen beträchtlichen Umfang an verschiedenen Fällen aufweisen, ist es regelmässig so, daß neben ganz wenigen interessanten Fällen nur noch triviale Fälle und "analog wie ..." - Fälle auftreten.

Stellt man selbst einen solchen Beweis auf, kann man leicht einen der wichtigen Fälle übersehen oder sich mit dem Überprüfen stets gleicher Fälle langweilen.

Die vorliegende Arbeit nutzt die Klassifizierung der Termbildungsregeln in Einführungs- und Beseitigungsregeln (wobei die Beseitigungen noch in kritische und nicht kritische geteilt werden), um die Fülle der Fallunterscheidungen in den Griff zu bekommen.

Der intuitive Hintergrund dieser Klassifizierung stammt von der Anwendung auf natürliche Herleitungen. Die logischen Schlußregeln können dort so aufgeteilt werden: - Regeln, die die Einführung einer logischen Konstanten erlauben (d.h. die Konklusion hat diese Konstante als äußerstes Symbol)) und - Regeln, die die Beseitigung

einer logischen Konstante erlauben (d.h. eine Prämisse, der Hauptteil, hat die Konstante als äußerstes Symbol) .

Die aus dieser Einteilung sich ergebenden Zwänge lassen noch Raum für die verschiedenartigsten Kalküle. Auf der anderen Seite scheint diese Einteilung wirklich ein Schlüssel für die Gültigkeit eines Church-Rosser Theorems zu sein. Es ergeben sich natürliche Begriffsbildungen und Beweisschritte.

Ist einmal klar, wie die "allgemeinen" Terme und das "allgemeine" Church-Rosser Theorem aussehen soll, so bleibt das Problem der Notation.

Zunächst braucht man zur Definition "allgemeiner" Terme/Reduktionen Metabegriffe (etwa Variable für Terme ect...). Es fällt also ein größerer Begriffsaufwand an Metasprache an, um die Regeln der Termbildung oder Reduktion formulieren zu können. Ein gewisser Standard ist hier leider noch nicht vorhanden. Die im nächsten Abschnitt gegebene Formulierung lehnt sich an die Arbeit "A General Church-Rosser Theorem" (1978) von Peter Aczel [1] an.

Der entstehende Kalkül ist eine Verallgemeinerung des Lambdakalküls. Er eignet sich zur Beschreibung aller Arten von Termkalkülen, die sich des Mechanismus der Anwendung und invers dazu der Einsetzung für Variablen bedient.

In den folgenden Beweisen selbst ist zum Glück dieser Aufwand nicht mehr so sichtbar wie bei den Definitionen, ja, die stärkere Allgemeinheit führt - wie oft in der Mathematik - sogar zu übersichtlicheren Beweisen.

Der zweite Teil des Notationsproblems ergibt sich aus dem Aufbau von Termen aus unterschiedlich vielen Teiltermen, was zum Hantieren mit vielen Indices zwingt. Die Standardmethode, dies zu umgehen, ist die Vektorschreibweise: $\vec{a} = (a_1, a_2, \dots)$. Um auch noch die Pfeile zu sparen, kann man sich auch darauf beschränken, die Beweise für eine feste (möglichst kleine) Anzahl von Teiltermen auszuführen. Da an Allgemeinheit dadurch im vorliegenden Fall wirklich nichts verloren geht, die Lesbarkeit aber sehr gefördert wird, habe ich diese Form der Darstellung gewählt.

2. Definition

Die folgenden abstrakten Begriffsbildungen sind im wesentlichen durch die Anwendung auf λ -Kalküle und Kalküle des natürlichen Schließens motiviert.

Zusätzlich zu den am Ende des Kapitels gegebenen Beispielen sind zum Zweck der leichteren Lesbarkeit deshalb den Definitionen bereits Beispiele zur Erläuterung beigelegt.

a) Termaufbau

FORMEN

Eine Form ist ein Symbol $K, B, E, \dots$ (auch mit Indices) zusammen mit seiner Wertigkeit $(k_1, k_2, \dots)$ $k_i \in \mathbb{N}$. Die Wertigkeit ist ein n -Tupel natürlicher Zahlen (einschließlich der 0), wobei $0 \leq n \leq \omega$. (Sowohl leere als auch unendliche Tupel sind erlaubt; die Beschränkung $\leq \omega$ ist nicht wesentlich).

n heißt die Länge der Form.

Die intuitive Bedeutung der Wertigkeit erschließt sich aus den Termbildungsregeln als Angabe darüber wieviele "Argumente" eine Form hat und wieviele Variable wo gebunden werden.

Formen der Länge 0 sind einfach Konstanten.

TERME

Terme $(t, s, r, \dots)$ eventuell indiziert) werden aus Formen und Variablen $(x, y, z, \dots)$ ebenso mit Indices) gebildet nach den Regeln:

1) Jede Variable ist ein Term.

2) Ist K eine Form mit Wertigkeit $(k_1, k_2, \dots)$,

$x_j^1, \dots, x_j^{k_j}$ Listen von Variablen (im folgenden kurz $\vec{x}_j$) und

t_j Terme für $j=1, 2, \dots$

so ist $K((x_1)t_1, (x_2)t_2, \dots)$ ein Term.

Freies und gebundenes Auftreten von Variablen ist wie üblich erklärt.

Eine Variable x_j^i , die in t_j frei ist, ist in $K(\vec{x}_1)t_1, (\vec{x}_2)t_2, \dots$

gebunden. Terme, die sich nur durch gebundene Variable unterscheiden, werden identifiziert.

Beispiel a) Abstraktion und Anwendung

Im reinen λ -Kalkül hat man die Form λ mit Wertigkeit (1) und die Form App (Application) mit Wertigkeit (0,0) .

Terme sind dann:

$\lambda((x)t)$ anschaulich die Funktion $s \mapsto t_x[s]$

App(t,s) anschaulich t angewandt auf s. In der üblichen Schreibweise ts .

METATERME

Metaterme ($H, H', H_i, R_i, \dots$) werden aufgebaut wie Terme.

Zusätzlich sind aber Metavariablen ($X_i, Y_i, Z_i, S_i, T_i, \dots$) zugelassen. Diese werden beim Aufbau der Metaterme behandelt wie Formen mit einer Wertigkeit (0,0,0,...) .

Metaterme werden zur Formulierung etwa der Kontraktionsvorschriften benutzt. Um zu sagen (Vergleiche Beispiel a) oben), daß :

$\text{App}(\lambda((x)t), s) \vdash t_x[s]$ für beliebige Terme t und s , verwendet man Metavariablen T mit Wertigkeit (0) und S mit Wertigkeit (0) und schreibt:

$\text{App}(\lambda((x)T(x)), S) \vdash T(S)$.

Die Metavariablen T steht hier also für Terme mit einer ausgezeichneten freien Variablen (Wertigkeit von T ist (0)). Substitutionen für diese Variablen im Term lassen sich dadurch anzeigen.

Entsprechend dieser Verwendung ist im folgenden die Einsetzung definiert.

EINSETZUNG

Sind t und s Terme (Metaterme) und x eine Variable, so bezeichnet $t_x[s]$ den Term (Metaterm), der entsteht, wenn man die Variable x überall, wo sie in t frei vorkommt, durch s ersetzt. Durch vorheriges Umbenennen gebundener Variablen in t ist dabei sicher zu stellen, daß Variable, die in s frei sind, nicht in $t_x[s]$ gebunden werden.

Ist H ein Metaterm, der die Metavariablen Z mit einer Wertigkeit $(0,0,0,\dots)$ der Länge n enthält, t ein Term und $\vec{x}$ eine Liste $x_1, x_2, \dots$ von n Variablen, so bezeichnet $H_Z[(\vec{x})t]$ denjenigen Metaterm, der entsteht, wenn man in H jeden Teilterm $Z(T_1, T_2, \dots)$ ersetzt durch $t_{x_1, x_2, \dots}[T_1, T_2, \dots]$ (simultane Substitution).

SPEZIALFALL EINES METATERMES

Es sei H ein Metaterm und $Z_1, Z_2, \dots$ seien alle in H vorkommenden Metavariablen, Z_i mit Wertigkeit $(0,0,\dots)$ der Länge n_i .

Weiter seien t_i Terme und x_i Listen von Variablen der Länge n_i . $H_{Z_1, Z_2, \dots}[(\vec{x}_1)t_1, (\vec{x}_2)t_2, \dots]$ ist dann ein Term, ein sogenannter Spezialfall des Metatermes H .

Sind mehrere Metaterme gegeben, so erhält man daraus konsistente Spezialfälle, wenn man gleiche Metavariablen in allen Metatermen durch gleiche Terme und Variablenlisten ersetzt.

KONTRAKTIONSSCHEMA

Ein Kontraktionsschema besteht aus abzählbar vielen Formen unterteilt in Einführungsformen (meist mit E bezeichnet)

mit Wertigkeit $(k_1, k_2, \dots)$

und Beseitigungsformen (meist mit B bezeichnet)

mit Wertigkeit $(0, n_1, n_2, \dots)$.

Weiter enthält es für jede Kombination einer Beseitigungsform B und einer Einführungsform E eine zugehörige Kontraktionsvorschrift, d.h. ein Paar H, H' von Metatermen ohne freie Variable von der Form $H = B(E((x_1)(T_1(x_1)), (x_2)(T_2(x_2)), \dots), (y_1)(S_1(y_1)), (y_2)(S_2(y_2)), \dots))$ (die T_i und S_i sind dabei Metavariablen mit passender Wertigkeit) und H' beliebig aufgebaut aus den T_i , den S_i , der Form B

und gebundenen Variablen (nicht aber aus freien Variablen oder anderen Formen (etwa E)).

Statt Beseitigungsform und Einführungsform wird der Kürze halber auch nur von Einführung und Beseitigung gesprochen.

Das erste Argument einer Beseitigung, der sogenannte Hauptteil, ist ausgezeichnet. Zum einen werden dort keine Variablen gebunden und zum andern ist es ausschlaggebend für Kontraktionen.

Beispiel b) Konjunktion

Als Beispiel soll die Konjunktion im Kalkül für natürliches Schließen dienen. Es besteht aus einer

Einführung: $\wedge^+$ mit Wertigkeit (0,0) Regel: $(\wedge^+) \frac{A \quad B}{A \wedge B}$

und zwei

Beseitigungen: $\wedge_l^-, \wedge_r^-$ mit Wertigkeit (0)

Regel: $(\wedge_r^-) \frac{A \wedge B}{A} \quad (\wedge_l^-) \frac{A \wedge B}{B}$

die Kontraktionsvorschriften lauten:

$H = \wedge_r^-(\wedge^+(T_1, T_2))$, $T_1 = H'$ also: $(\wedge^+) \frac{\vdots A \quad \vdots B}{(\wedge_r^-) \frac{A \wedge B}{A}} \Rightarrow \vdots A$

$H = \wedge_l^-(\wedge^+(T_1, T_2))$, $T_2 = H'$ also: $(\wedge^+) \frac{\vdots A \quad \vdots B}{(\wedge_l^-) \frac{A \wedge B}{B}} \Rightarrow \vdots B$

Die Regel ist nicht kritisch.

KRITISCHE BESEITIGUNG

Eine Kontraktionsvorschrift heit kritisch, wenn

(Bezeichnungen wie unter KONTRAKTIONSSCHEMA) :

H' die Gestalt hat $B(R, (\bar{y}_1^*)(S_{i_1}(\bar{y}_1^*)), (\bar{y}_{i_2}^*)(S_{i_2}(\bar{y}_{i_2}^*)), \dots)$

wobei $\{i_1, i_2, \dots\} \subset \{1, 2, \dots\}$ und R ein Metaterm

ist, der nur aus den T_i und gebundenen Variablen aufgebaut ist

oder

H' die Gestalt hat $S_i(R_1, \dots, R_{k_i})$,

wobei $i \in \{1, 2, \dots\}$ fest und die R_j fr $j = 1, \dots, k_i$ Metaterme sind, die nur die T_i und gebundenen Variablen enthalten.

Eine Beseitigung B heit kritisch, wenn fr jede Einfhrung E des Kontraktionsschemas die zugehrige Kontraktionsvorschrift kritisch ist. Sonst heit B unkritisch.

Beispiel c) Ziffern

Ein Beispiel fr eine kritische Beseitigung bietet die "Rekursion" mit unendlichen Termen.

Fr jedes $n \in \mathbb{N}$ die Form $\underline{n}$ (Einfhrung) mit Wertigkeit $()$

und die Form $\langle \rangle$ (Beseitigung) mit Wertigkeit $(0, 0, 0, \dots)$.

Die Terme $\langle \rangle(t, s_0, s_1, \dots)$ schreibt man gewhnlich $\langle s_1 \rangle t$.

(Auswertung der Funktion $i \mapsto s_i$ an der Stelle t)

Die Kontraktion $\langle s_1 \rangle \underline{n} \mapsto s_n$ wird durch die Kontraktionsvorschrift:

"Fr jedes $n \in \mathbb{N}$ $H = \langle \rangle(\underline{n}, s_0, s_1, s_2, \dots)$, $s_n = H'$ "

beschrieben. Sie ist kritisch nach der zweiten Bedingung.

Kritische Regeln nach der ersten Bedingung sind seltener.

Als Beispiel diene eine Modifikation von Beispiel c) :

Beispiel d) Null und Nachfolger

Terme werden aufgebaut mit den Formen:

Einfhrungen : 0 (Null) mit Wertigkeit $()$ und

$\mathcal{Y}$ (Nachfolger) mit Wertigkeit (0)

Beseitigung : $\langle \rangle$ (Auswertung) mit Wertigkeit $(0, 0, 0, \dots)$

Die Kontraktionsvorschriften lauten hier:

$\langle \rangle(0, s_0, s_1, s_2, \dots)$, s_0

und $\langle \rangle(\mathcal{Y}(t), s_0, s_1, s_2, \dots)$, $\langle \rangle(t, s_1, s_2, \dots)$

NORM

Auf den Termen wird eine Norm $\| \cdot \|$ definiert durch:

$\|x\| = 0$ für jede Variable x .

$\|E((\vec{x}_1)t_1, (\vec{x}_2)t_2, \dots)\| = \sup_i \{\|t_i\|\} + 1$
für jede Einführungsform E .

$\|B(s, (\vec{x}_1)t_1, (\vec{x}_2)t_2, \dots)\| = \sup_i \{\|t_i\|\} + \|s\| + 1$
für jede Beseitigungsform B .

b) Modifikationen

In vielen Fällen liegt ein Termkalkül vor, der gegenüber dem oben Beschriebenen in der einen oder anderen Weise modifiziert ist. Die nächstliegende Modifikation ist die Hinzunahme einer Typstruktur. Dies wird zwangsläufig nötig auf dem Wege zu einem Normalisierungssatz, da -wie man weiß- ein Normalisierungssatz schon beim einfachen Lambda-kalkül nur für einen getypten Kalkül gilt.

Zum zweiten folgt aus dem Church-Rosser Theorem für einen ungetypten Kalkül nicht unmittelbar das Theorem für den getypten. (Man muß sich noch vergewissern, daß in einem getypten Kalkül bei jedem Reduktionsschritt ein typenrichtig aufgebauter Term in einen ebensolchen überführt wird.)

Noch stärker mit Information angereichert sind etwa Kalküle für natürliche Herleitungen. Der einfache Kalkül beschreibt dann nur die Grobstruktur eines Herleitungsbaumes, nämlich welche Regel an welchem Knoten angewendet wurde und die Variablen, die dabei eine Rolle spielen.

Da Terme mit Typen in der vorliegenden Arbeit selbst gebraucht werden, sind sie im folgenden ausführlich behandelt.

Die Behandlung natürlicher Herleitungen dagegen wird nur angedeutet.

Die folgenden Definitionen sind wie gewohnt:

TYPSYMBOL

Ein Typsymbol ist ein Symbol $(+, \rightarrow, *, o, \dots)$ mit einer Stelligkeit $k \in \mathbb{N}$. (kurz: k -stelliges Typsymbol)

TYP

Typen sind induktiv definiert durch:

Sind $\mathcal{G}_1, \dots, \mathcal{G}_k$ Typen und $*$ ein k -stelliges Typsymbol, so ist $* \mathcal{G}_1 \dots \mathcal{G}_k$ ein Typ.

Bei zweistelligen Typsymbolen wird auch die Schreibweise $\mathcal{G}_1 * \mathcal{G}_2$ verwendet.

TYPNIVEAU

Das Typniveau $n(\mathcal{G})$ eines Typs $\mathcal{G}$ wird induktiv definiert durch:

$n(* \mathcal{G}_1 \dots \mathcal{G}_k) = \max \{ n(\mathcal{G}_1), \dots, n(\mathcal{G}_k) \} + 1$ (Es sei $\max \{ \} = 0$).

Unter dem Typniveau $n(t)$ eines Termes t verstehe ich das Typniveau $n(\mathcal{G})$ des Typs $\mathcal{G}$ von t .

Beispiel a) (Fortsetzung)

Im λ -Kalkül aus Beispiel a) hat man die Typsymbole o nullstellig und $\rightarrow$ zweistellig, aus denen man Typen o , $o \rightarrow o$, $(o \rightarrow o) \rightarrow o$ etc. aufbaut.

Das Typniveau reflektiert das gewöhnliche Typniveau nur beschränkt, da $n(\mathcal{G} \rightarrow \mathcal{r}) = \max \{ n(\mathcal{G}), n(\mathcal{r}) \} + 1$ nicht $\max \{ n(\mathcal{G}) + 1, n(\mathcal{r}) \}$.

TYPAUSDRUCK

Für Typen werden $\mathcal{G}, \mathcal{r}, \mu, \dots$ als Metavariablen verwendet. Aus ihnen und Typsymbolen lassen sich Ausdrücke aufbauen, wie etwa $* \mathcal{G} \mathcal{r}$, die im folgenden als Typausdrücke bezeichnet werden.

Als Mitteilungszeichen für Typausdrücke werden ebenfalls die Buchstaben $\mathcal{G}, \mathcal{r}, \mu, \dots$ verwendet.

Ebenso wie im Zusammenhang mit Metatermen, Metavariablen und Termen der Begriff (konsistente) Spezialfälle definiert wurde, sei bei Typausdrücken, Typvariablen und Typen dieser Begriff völlig analog definiert.

Ergänzend ist eine (konsistente) Verallgemeinerung definiert durch:
Ersetzt man in einem Typausdruck eine Typvariable überall, wo sie vorkommt, durch einen Typausdruck, so erhält man wieder einen Typausdruck - eine sogenannte Verallgemeinerung des Typausdruckes.
Ersetzt man in mehreren Typausdrücken die gleiche Variable durch den gleichen Typausdruck, erhält man konsistente Verallgemeinerungen.

GETYPTE VARIABLE

Eine getypte Variable x^{σ} ist eine Variable zusammen mit einem Typ σ .
Der Typindex wird oft weggelassen, wenn der Typ unwichtig oder aus dem Zusammenhang klar ist.

GETYPTE FORM

Eine getypte Form ist eine Form K mit Wertigkeit $(k_1, k_2, \dots)$ zusammen mit einem Typschema der Form:

$$((\sigma_1^1, \sigma_2^1, \dots, \sigma_{k_1}^1) \sigma^1, (\sigma_1^2, \dots) \sigma^2, \dots) \sigma$$

Dabei sind die vorkommenden $\sigma_j^i, \sigma^i, \sigma$ beliebige Typausdrücke mit der Einschränkung, daß im letzten Typausdruck σ nur solche Typvariablen vorkommen dürfen, die bereits in einem weiter links stehenden Typausdruck vorkommen.

Dieser letzte Typausdruck wird auch etwas ungenau als Typ der Form bezeichnet.

Der Sinn des Typschema ergibt sich einfach aus der Notwendigkeit, bei den Termbildungsregel angeben zu müssen, wie sich der Typ von $K(\dots)$ aus den Typen der Teilterme und der gebundenen Variablen ergibt und welche Bedingungen an diese Typen zu stellen sind. Nämlich:

GETYPTER TERM

Getypte Terme sind induktiv definiert durch:

- 1) Getypte Variable x^G vom Typ G sind getypte Terme vom Typ G .
- 2) Ist K eine getypte Form mit Typschema $((G_1^1, \dots)G_1^1, \dots)G$ und sind μ_j^1, μ^1, μ konsistente Spezialfälle der G_j^1, G^1, G , so ist für beliebige getypte Terme t^1 vom Typ μ^1 und Variable x_j^1 vom Typ μ_j^1 $K((x_1^1, \dots, x_{k_1}^1)t^1, \dots)$ ein getypter Term vom Typ μ .

Beispiel a) (Fortsetzung).

Erweitert man Beispiel a) um Typen so erhält man:

für λ das Typschema $((G_1^1)G_1^1)G_1^1 \rightarrow G^1$

für App das Typschema $(G \rightarrow \tau, G)\tau$,

so daß $\lambda((x^G)t^\tau)$ vom Typ $G \rightarrow \tau$ und $\text{App}(t^{G \rightarrow \tau}, s^G)$ vom Typ τ ist.

Die Verallgemeinerung auf Metaterme (siehe unten) ist unmittelbar:

$$\text{App}(\lambda((x^G)t^\tau(x^G)), s^G) \vdash t^\tau(s^G)$$

beschreibt die Kontraktion für beliebige Terme t vom beliebigen Typ τ , eine Variable vom beliebigen Typ G und Terme s vom Typ G .

Die Terme werden eben durch formale Termvariable (Metavariable) und die Typen durch formale Typvariable vertreten.

GETYPTE METAVARIABLE

Getypte Metavariable werden definiert wie getypte Formen mit Wertigkeit $(0, \dots, 0)$.

GETYPTE METATERME

Getypte Metaterme werden analog zu den getypten Termen definiert, wobei allerdings statt getypten Formen auch getypte Metavariable zulässig sind.

Weiter ist einem Metaterm kein Typ, sondern ein Typausdruck zugeordnet. Dieser wird auch als Typ des Metatermes bezeichnet.

Konsequenterweise läßt man auch Variable zu, denen kein Typ, sondern ein Typausdruck zugeordnet ist.

Im Einzelnen:

- 1) Variable x^G mit Typausdruck G sind getypte Metaterme vom Typ G .

2) Ist K eine getypte Form oder getypte Metavariablen mit

Typschema $((\sigma_1^1, \dots, \sigma_k^1) \sigma^1, \dots) \sigma$

und sind μ_j^1, μ^1, μ konsistente Verallgemeinerungen der $\sigma_j^1, \sigma^1, \sigma$ so ist für beliebige getypte Metaterme T^1 vom Typ μ_j^1

und Variablen x_j^1 mit Typausdruck μ_j^1 $K((x_1^1, \dots, x_k^1) T^1, \dots)$

ein getypter Metaterm vom Typ μ .

EINSETZUNG IM GETYPTEN KALKÜL

Ist t ein Term mit Typ σ , s ein Term vom Typ τ und x

eine Variable vom Typ τ , dann ist $t_x[s]$ wieder ein getypter Term vom gleichen Typ wie t . Entscheidend hängt diese Tatsache davon ab, daß die Variable und der eingesetzte Term den gleichen Typ haben.

Für Einsetzungen bei Metatermen gilt das Entsprechende, falls Metavariablen und eingesetzter Metaterm den gleichen Typausdruck haben.

SPEZIALFALL EINES METATERMES

Spezialfälle und konsistente Spezialfälle sind wie im ungetypten Kalkül definiert, lediglich ist bei Einsetzungen darauf zu achten, daß die Typen der eingesetzten Terme und Variablen den Typschemata der ersetzten Metavariablen entsprechen.

GETYPTES KONTRAKTIONSSCHEMA

Ein Kontraktionsschema ist getypt, wenn zum einen seine Formen getypte Formen und die Metaterme H, H' der Kontraktionsvorschriften getypte Metaterme sind und zum andern die folgenden Bedingungen erfüllt sind:

1) Ist E eine Einführung und B eine Beseitigung des Kontraktionsschemas so hat das Typschema von E die Gestalt

$((\sigma_1^1, \sigma_2^1, \dots) \sigma^1, \dots) * \tau_1 \tau_2 \dots$ und

das Typschema der Beseitigungsform B die Gestalt

$(* \tau_1 \tau_2 \dots, (\mu_1^1, \mu_2^1, \dots) \mu^1, \dots) \mu$.

D.h. der Hauptteil der Beseitigung verlangt einen Typ, den man mit einer Einführung erzeugen kann.

2) Die Metaterme H, H' der Kontraktionsvorschriften haben je den gleichen Typ.

Dies ist wichtig, damit bei der Reduktion typenrichtig aufgebaute Terme in typenrichtig aufgebaute Terme übergehen.

KRITISCHE FORM

Die Bedingungen für kritische Beseitigungen sind die selben wie im ungetypten Kalkül. Außerdem muß das Typschema folgende Gestalt besitzen:

$$(\sigma^0, (\sigma_0^1, \sigma_1^1, \dots) \sigma, (\sigma_0^2, \sigma_1^2, \dots) \sigma, (\sigma_0^3, \sigma_1^3, \dots) \sigma, \dots) \sigma$$

dh. die Typen von 2, 3, 4, ...-tem Argument und des Termes selbst sind gleich.

c) Reduktion

Auf den Termen werden im folgenden verschiedene Reduktionsrelationen definiert, die mit den Zeichen $\vdash$ und $\gg$ bezeichnet werden.

Es sind Ordnungsrelationen auf den Termen.

Je nach dem Satz, den man beweisen möchte, ist die Reduktionsrelation als eine geeignete Modifikation der folgenden Relation definiert:

1) "Reflexivität"

$t \vdash t$ für jeden Term t .

2) "Echte Reduktion"

Ist H, H' eine Kontraktionsvorschrift des Kalküls und sind t, t' konsistente Spezialfälle von H, H' so $t \vdash t'$.

3) "Permutative Reduktion"

Sind B, B' Beseitigungen, wovon B' kritisch ist, so gilt mit geeigneten Termen s_i, t_i und Variablen x_i, y_i :

$$B'(s_0, (\vec{x}_1) s_1, \dots), (\vec{y}_1) t_1, \dots \vdash B(s_0, (\vec{x}_1) B(s_1, (\vec{y}_1) t_1, \dots), \dots)$$

4) "Verträglichkeit"

Gilt $t_0 \vdash t'_0$, $t_1 \vdash t'_1$, ... so folgt für jede Form K

$$K((\vec{x}_0) t_0, (\vec{x}_1) t_1, \dots) \vdash K((\vec{x}_0) t'_0, (\vec{x}_1) t'_1, \dots)$$

5) "Transitivität"

Gilt $t \vdash s$ und $s \vdash t'$ so gilt $t \vdash t'$.

Die Numerierung der Regeln wird im folgenden immer der obigen entsprechen. Werden einzelne Regeln (etwa die Vertraglichkeit) aus technischen Gründen aufgespalten in mehrere Teilregeln so ist die Numerierung 4), 4'), 4'')

Sind einige Reduktionen nicht vorhanden, so wird die Numerierung nicht fortlaufend.

3. -- weitere Beispiele --

e) Paarbildung und Projektion

Zweistelliges Typsymbol: •

Einführung: $[]$ mit Wertigkeit (0,0)

und Typschema $(\sigma, \tau) \sigma \cdot \tau$

Beseitigung: p mit Wertigkeit (0)

und Typschema $(\sigma \cdot \tau) \sigma$

q mit Wertigkeit (0)

und Typschema $(\sigma \cdot \tau) \tau$

Kontraktionsvorschriften:

$H = p([T_1, T_2])$, $T_1 = H'$

$H = q([T_1, T_2])$, $T_2 = H'$

Die Formen p und q sind nicht kritisch.

f) Injektion und Fallunterscheidung

Zweistelliges Typsymbol: +

für jeden Typ $\mathcal{G}$ die

Einführung: $i_{\mathcal{G}}$ mit Wertigkeit (0)

und Typschema $(\tau) \tau + \mathcal{G}$

$j_{\mathcal{G}}$ mit Wertigkeit (0)

und Typschema $(\tau) \mathcal{G} + \tau$

Beseitigung: D mit Wertigkeit (0,1,1)

und Typschema $(\mathcal{G} + \tau, (\mathcal{G})_{\mu}, (\tau)_{\mu})_{\mu}$

Kontraktionsvorschriften:

für jeden Typ

$$H = D(i_{\mathcal{G}}(T^{\tau}), (x^{\tau})(S_1^{\tau}(x^{\tau})), (y^{\mathcal{G}})(S_2^{\tau}(y^{\mathcal{G}}))) \quad , \quad S_1^{\tau}(T^{\tau}) = H'$$

$$H = D(j_{\mathcal{G}}(T^{\tau}), (x^{\mathcal{G}})(S_1^{\tau}(x^{\mathcal{G}})), (y^{\tau})(S_2^{\tau}(y^{\tau}))) \quad , \quad S_2^{\tau}(T^{\tau}) = H'$$

Die Beseitigung D ist kritisch.

g) Null, Nachfolger und Rekursion

Nullstelliges Typsymbol: o

Einführungen: O mit Wertigkeit ()

und Typschema () o

$\mathcal{Y}$ mit Wertigkeit (0)

und Typschema (o)o

Beseitigungs: R mit Wertigkeit (0,0,2)

und Typschema $(o, \mu, (\mu, o)_{\mu})_{\mu}$

Kontraktionsvorschriften:

$$H = R(O, S_1^{\mu}, (x^{\mu}, y^o)(S_2^{\mu}(x^{\mu}, y^o))) \quad , \quad S_1^{\mu} = H'$$

$$H = R(\mathcal{Y}(T^o), S_1^{\mu}, (x^{\mu}, y^o)(S_2^{\mu}(x^{\mu}, y^o))), \\ S_2^{\mu}(R(T^o, S_1^{\mu}, (x^{\mu}, y^o)(S_2^{\mu}(x^{\mu}, y^o))), T^o) = H'$$

Diese Beseitigung ist nicht kritisch.

NATÜRLICHE HERLEITUNGEN

Ebenso, wie bei den vorigen Beispielen in einem getypten Kalkül Variable von verschiedenem Typ auf den gleichen ungetypten Vorrat von Variablen abgebildet werden, stehen bei den natürlichen Herleitungen die Variablen für Annahmen beliebiger Formeln.

Bei den Reduktionen werden diese Variablen durch Terme ersetzt, was der Substitution einer Annahme durch die entsprechende Herleitung entspricht.

Bei der Codierung von Herleitungsbäumen als Terme entsprechen nämlich die Formen den einzelnen Regelanwendungen im Baum.

Der codierende Term enthält so die Struktur des Baumes, d.h. für jeden Knoten die dort angewendete Regel.

Der Herleitungsbaum ist nun durch den zugeordneten Term zusammen mit der Bedeutung der freien Variablen im Term eindeutig bestimmt.

"Bedeutung der freien Variablen" meint hier: Jede Variable steht für eine Annahme. Die Variable selbst gibt aber keinen Aufschluß mehr darüber, um welche Annahme es sich handelt. Die Zuordnung von Variablen zu Annahmen ist oben mit Bedeutung der Variablen gemeint.

h) einfache natürliche Herleitungen

Im folgenden wird ein einfacher Kalkül für natürliche Herleitungen zunächst informell beschrieben. Danach wird seine Einbettung in den allgemeinen Rahmen angegeben.

Der Kalkül enthält zunächst Objektterme mit einfachen Typen.

Die Terme werden aufgebaut aus Variablen, durch Anwendung.

Einfache Typen werden aufgebaut aus 0 (nullstelliges Typsymbol) und (,) (zweistelliges Typsymbol).

Jeder dieser Terme werde mit seiner Normalform identifiziert. (Wichtig für spätere Erweiterungen des Termvorrates.)

Die Axiome und Regeln des Kalküls sind:

(-) $t = t$ für jeden Objektterm t

(T)
$$\frac{t = s \quad s = u}{t = u} \quad \text{Transitivität}$$

$(\lambda_r^-) \frac{t = s}{ut = us} \quad (\lambda_l^-) \frac{t = s}{tu = su}$

$$\begin{array}{lcl}
 (\forall^-) \frac{A}{\forall x:A} & (\forall^-) \frac{\forall x:A}{A[t]} & \text{Allquantor} \\
 (\rightarrow^-) \frac{A \rightarrow B \quad A}{B} & (\rightarrow^-) \frac{B}{A \rightarrow B} & \text{Implikation} \\
 (\perp^-) \frac{\perp}{A} & & \text{Falsum}
 \end{array}$$

Die Kontraktionen sind:

gültig:

$$\begin{array}{ccc}
 \begin{array}{c} x \\ \vdots \\ A \\ (\forall^-) \frac{A}{\forall x:A} \\ (\forall^-) \frac{\forall x:A}{A[t]} \end{array} & \vdash & \begin{array}{c} A \\ \vdots \\ E \\ (\rightarrow^-) \frac{E}{A \rightarrow B} \\ (\rightarrow^-) \frac{A \rightarrow B}{B} \end{array} \quad \vdash \quad \begin{array}{c} A \\ \vdots \\ A \\ \vdots \\ B \end{array}
 \end{array}$$

und permutativ:

$$\begin{array}{ccc}
 \begin{array}{c} (T) \frac{t = u \quad u = v}{t = v} \\ (\lambda_L^-) \frac{t = u}{tv = uv} \end{array} & \vdash & \begin{array}{c} (\lambda_L^-) \frac{t = u}{tv = uv} \quad (\lambda_L^-) \frac{u = v}{uv = uv} \\ (T) \frac{tv = uv}{tv = uv} \end{array}
 \end{array}$$

Bei der $\forall$ -Kontraktion wird für eine Objektvariable x ein Objektterm t substituiert. (x kann dabei nicht frei in einer Annahme vorkommen, sondern muß durch eine der Regeln $(\perp^-)$, (λ_L^-) , (λ_V^-) , $(=)$ oder $(\forall^-)$ in die Herleitung gekommen sein.)

Ähnlich bei der $\rightarrow$ -Kontraktion. Hier wird für eine Annahme A eine Herleitung dieser Formel substituiert.

Man braucht, um diese Substitutionen beschreiben zu können also einen Kalkül, der gleichermaßen Variable für Objektterme wie für Herleitungen enthält.

Weiter müssen alle Regeln des Kalküls invariant unter Substitutionen für solche Variablen sein.

Daraus ergibt sich zwangsläufig, daß die Herleitungsregeln $(\forall^-)$, (λ_V^-) , (λ_L^-) und $(=)$ als zusätzliche Prämisse den Term t bzw u erhalten müssen, der in der Konklusion sonst neu auftaucht und die Invarianz unter Substitutionen verletzen würde.

Um solche Prämissen gewinnen zu können, wird eine weitere Regel hinzugenommen:

(Term) t für jeden Term t .

Die Typen im Kalkül bestehen nun aus den einfachen Typen (siehe oben) und außerdem den Typsymbolen

- $-$ einstelliges Typsymbol
- $\vee$ zweistelliges Typsymbol
- $\rightarrow$ zweistelliges Typsymbol
- $\perp$ einstelliges Typsymbol

und aus technischen Gründen für die Transitivität das Typsymbol tr nullstelliges Typsymbol

Die zu den Regeln gehörenden Formen werden beschrieben durch:

	Form	Wertigkeit	Typschema
a)	$-$	(0)	$(\sigma) = \sigma$
b)	tr	(0,0,0)	$(tr, =\sigma, =\sigma) = \sigma$
c)	λ_L^-	(0,0)	$(=(\sigma, \tau), \sigma) = \tau$
d)	λ_r^-	(0,0)	$((\sigma, \tau), =\sigma) = \tau$
e)	$\vee^+$	(1)	$((\sigma) \tau) \vee \sigma \tau$
f)	$\vee^-$	(0,0)	$(\vee \sigma \tau, \sigma) \tau$
g)	$\rightarrow^+$	(1)	$((\sigma) \tau) \sigma \rightarrow \tau$
h)	$\rightarrow^-$	(0,0)	$(\sigma \rightarrow \tau, \sigma) \tau$
i)	$\perp_A$	(0,0,0,...)	$(\perp(\sigma), \sigma_0, \sigma_1, \dots) \sigma$
j)	$Term_\sigma$	(0,0,0,...)	$(\sigma_0, \sigma_1, \sigma_2, \dots) \sigma$

Kontraktionsschemata:

Die Regeln a), b), c), d), i) und j) sind jeweils Kontraktionsschemata, die nur eine einzige Form enthalten also keine Kontraktionsvorschriften erfordern. Die Regeln a) und j) werden als Einführungen klassifiziert und die Regeln b), c), d) und i) als Beseitigungen.

Aufgrund des Typschemas ist nur b) eine kritische Beseitigung.

Die Transitivität hat als kritische Regel, wie man aus der informellen Beschreibung der Kontraktionen sieht, nur Nebenteile und keinen Hauptteil. Um dies im gegebenen Formalismus beschreiben zu können, fügt man künstlich einen Hauptteil hinzu, der den Typ tr haben muß. Für diesen

Typ gibt es keine Einführungen. Der Hauptteil kann einfach durch eine Variable vom Typ τ gebildet werden.

Die Regeln Term_t und $\perp_A$ führen Terme b.z.w. Aussagen in die Herleitungen ein. Um diese Regeln invariant unter Substitutionen zu machen, haben sie Argumente vom Typ $\mathcal{G}_0, \mathcal{G}_1, \mathcal{G}_2, \dots$ bekommen. Diese Argumente sind die freien Variablen in der Konklusion, für die substituiert werden kann.

Eine Regel $(\perp) \frac{\perp}{x \approx x}$ wird also beschrieben durch $\perp_{x=x}(\vdots, x, y, y, \dots)$. Dabei sei eine beliebige Zuordnung zwischen den freien Variablen im Index der Regel und den Argumentstellen gegeben. Ungebrauchte Argumentstellen werden beliebig aufgefüllt. Analog wird $(\text{Term}) \ x(yz)$ durch $\text{Term}_{x_1(x_2x_3)}(x, y, z, w, w, w, \dots)$ oder $\text{Term}_{x_1x_2}(x, \text{Term}_{x_3x_4}(w, w, y, z, w, w, \dots), w, w, \dots)$ beschrieben.

Die Kontraktions schemata für e), f) und g), h) enthalten die Vorschriften:

$\nabla^-(\nabla^+((x)T(x)), S)$, $T(S)$ und $\rightarrow^-(\rightarrow^+((x)T(x)), S)$, $T(S)$ beide Regeln sind nicht kritisch.

Es ergeben sich so die obigen Kontraktionen. Die Normalisierungsbedingungen aus Kap III sind erfüllt und so folgt die Existenz und Eindeutigkeit der Normalformen für obigen Kalkül.

Erweiterungen:

Das Nächstliegende ist die Hinzunahme der Regel

$$(\lambda^+) \frac{t = s}{\lambda_{x.t} = \lambda_{x.s}}, \text{ Form } \lambda^+, \text{ Wertigkeit } (1) \text{ und}$$

Typschema $((\mathcal{G}), =\tau) = (\mathcal{G}, \tau)$. Zusammen mit Regel c) oben ergibt sich damit ein Kontraktions schema mit der Kontraktionsvorschrift

$$\lambda_e^-(\lambda^+((x)T(x)), S), T(S).$$

Hinzu gehört natürlich die Erweiterung des Bereichs der Objekterme um die λ -Abstraktion.

Weiter kann der Kalkül um die unten beschriebenen Regeln für Disjunktion und Konjunktion sowie ω -Regel und Existenzquantor erweitert werden.

Auch der Bereich der Objektterme läßt sich problemlos ausdehnen.

Etwa durch Terme $\underline{n}$ und $\langle s_i \rangle t$ zusammen mit den Regeln

$$(\langle \rangle \bar{e}) \quad \frac{t = s \dots u_i \dots}{\langle u_i \rangle t = \langle u_i \rangle s} \quad \text{und} \quad (\langle \rangle \bar{v}) \quad \frac{t \dots u_{i1} = u_{i2} \dots}{\langle u_{i1} \rangle t = \langle u_{i2} \rangle t}$$

Die Regel $\langle \rangle \bar{e}$ ist kritisch, was zu neuen permutativen Reduktionen führt.

Problematisch wird die Erweiterung, wenn die Verträglichkeit der Gleichheit mit Kontraktionen der Objektterme formuliert werden soll, wie in der Regel:

$$\frac{\begin{array}{c} x \\ \vdots \\ t = s \end{array}}{\lambda x. t = \lambda x. s} \quad u = v \quad \triangleright \quad \begin{array}{c} \vdots \\ u = v \\ \vdots \end{array} \quad t_x[u] = s_x[v]$$

Hier ist unklar, was unter der Substitution einer Gleichung für einen Term zu verstehen ist, obwohl natürlich, ausgehend vom Aufbau von t und s , eine solche Herleitung definierbar ist. Dies jedoch nicht durch Substitution alleine. (Transitivität wird zusätzlich erforderlich.)

Ein Weg, diese Schwierigkeit zu umgehen, ist der Ersatz der Gleichheit durch endliche nichtleere Mengen von Termen. Eine solche Termmenge steht für die Aussage "Alle Terme der Menge sind gleich"

Die λ -Regeln werden so zu

$$(\lambda^+) \frac{\{t_i\}}{\{\lambda x. t_i\}} \quad (\lambda^-) \frac{\{t_i\} \{s_j\}}{\{t_i s_j\}}$$

und die Transitivität zu $(T) \frac{\{t_i, u_j\} \quad \{u_j, s_k\}}{\{t_i, s_k\}}$

Diese Regeln umfassen sowohl den Fall von Gleichungen als auch von Einzeltermen.

Bei der Substitution einer Menge für eine Variable in einer Menge ist zu beachten, daß stets alle möglichen Kombinationen entstehen.

D.h.: $\{t_i\}_x [\{s_j\}] = \{(t_i)_x [s_j]\}$

Weitere Beispiele:

i) Disjunktion

Für jede Formel B die

Einführung: $\bigvee_B^+$ mit Wertigkeit (0) Regel: $(\bigvee_B^+) \frac{A}{A \vee B}$

und für jede Formel A die

Einführung: $\bigvee_A^+$ mit Wertigkeit (0) Regel: $(\bigvee_A^+) \frac{B}{A \vee B}$

Beseitigung: $\bigvee^-$ mit Wertigkeit (0,1,1)

Kontraktionsvorschrift:

$H = \bigvee^-(\bigvee_A^+(T), (x)S_1(x), (y)S_2(y))$, $S_2(T) = H'$ also:

$$\begin{array}{c}
 \begin{array}{c}
 \vdots \\
 \vdots \\
 \vdots \\
 \vdots \\
 \vdots
 \end{array}
 \end{array}
 \begin{array}{c}
 A \\
 \vdots \\
 \vdots \\
 \vdots \\
 \vdots
 \end{array}
 \begin{array}{c}
 B \\
 \vdots \\
 \vdots \\
 \vdots \\
 \vdots
 \end{array}
 \begin{array}{c}
 \vdots \\
 B \\
 \vdots \\
 \vdots \\
 \vdots
 \end{array}$$

$$(\bigvee^-) \frac{(\bigvee^+) \frac{A}{A \vee B} \quad B}{A \vee B} \quad C \quad C \quad C \quad \rightarrow \quad C$$

ebenso:

$H = \bigvee^-(\bigvee_B^+(T), (x)S_1(x), (y)S_2(y))$, $S_1(T) = H'$

Die Regel ist kritisch.

k) Existenzquantor

Einführung: $\exists^+$ mit Wertigkeit (0) Regel: $(\exists^+) \frac{A[s]}{\exists x:A}$

Beseitigung: $\exists^-$ mit Wertigkeit (0) Regel: $(\exists^-) \frac{\exists x:A}{A[s]}$

Kontraktionsvorschrift:

$H = \exists^-(\exists^+(T))$, $T = H'$ also:

Die Regel ist nicht kritisch.

$$\begin{array}{c} \vdots \\ (\exists^+) \frac{A[s]}{\exists x:A} \\ (\exists^-) \frac{\exists x:A}{A[s]} \end{array} \Rightarrow \begin{array}{c} \vdots \\ A[s] \end{array}$$

Implizit enthält die Herleitung von $\exists x:A$ stets den realisierenden Term s, obwohl er in der Formel selbst nicht zu sehen sein mag.

alternativ:

Einführung: $\exists^+$ mit Wertigkeit (0)

$$\text{Regel: } (\exists^+) \frac{A[s]}{\exists x:A}$$

Beseitigung: $\exists^-$ mit Wertigkeit (0,1) $A[x]$

$$\text{Regel: } (\exists^-) \frac{\exists x:A \quad \begin{array}{c} \vdots \\ C \end{array}}{C}$$

(Nebenbedingung: x nicht frei in C und $\exists x:A$.)

Kontraktionsvorschrift:

$H = \exists^-(\exists^+(T), (x)S(x))$, $S(T) = H'$

Die Regel ist kritisch.

l) ω -Regel

Einführung: ω^+ mit Wertigkeit (0,0,...) Regel: $(\omega^+) \frac{\dots A[i] \dots}{\forall x:A}$

Beseitigung: $\omega^{-,n}$ mit Wertigkeit (0) Regel: $(\omega^{-,n}) \frac{\forall x:A}{A[n]}$

Kontraktionsvorschrift:

$H = \omega^{-,n}(\omega^+(\dots, T_1, \dots))$, $T_n = H'$ also:

Die Regel ist nicht kritisch.

$$\begin{array}{c} \vdots \\ (\omega^+) \frac{\dots A[i] \dots}{\forall x:A} \\ (\omega^{-,n}) \frac{\forall x:A}{A[n]} \end{array} \Rightarrow \begin{array}{c} \vdots \\ A[n] \end{array}$$

Ein Kalkül, der sich besonders gut für Beispiele in den nachfolgenden Sätzen eignet, ist der Termkalkül, der aus den Formen von a) und c) besteht. (Siehe Seite 4 und 7)

Dieser Termkalkül wird im folgenden mit T_{∞} bezeichnet.

Es wird dabei eine vereinfachte Schreibweise benutzt:

App wird ganz weggelassen, ts statt $\text{App}(t,s)$,

und statt $\langle \rangle(t, u_0, u_1, \dots)$ schreibt man $\langle u_i \rangle t$.

Es lassen sich in diesem Kalkül praktisch alle Probleme exemplarisch vorführen.

II. Church-Rosser-Theorem

Das Resultat dieses Kapitels ist ein Church-Rosser Theorem für den in Kapitel I beschriebenen allgemeinen Termkalkül.

Nochmals betont sei, daß der vorliegende Beweis sowohl permutative Reduktionen als auch unendliche Terme erlaubt.

Die größere Allgemeinheit, die sich daraus ergibt, daß der Beweis nicht für einen speziellen Termkalkül, sondern für eine Vielzahl "ähnlicher" Kalküle geführt wird, führte nicht zu einem besonders komplizierten Beweis, sondern macht die wirklich entscheidenden Mechanismen erst sichtbar und führte so (nach Meinung des Autors) zu einem "einfachen" Beweis.

Für Kalküle mit unendlichen Termen genügen anscheinend streng finite Beweismittel zum Beweis des Church-Rosser Theorems nicht mehr (siehe etwa Girard [3]).

Der hier vorgestellte Beweis arbeitet mit transfiniten Induktion. Er geht in der Hauptidee auf Maß [5],[6] zurück.

Die Hauptschwierigkeit besteht bei einem Induktionsbeweis in der passenden Definition einer Einschrittreduktion. Dies gilt umso mehr, wenn permutative Reduktionen zugelassen sind.

Deshalb wird hier zuerst ein Reduktionskalkül $\alpha \vdash_n$ definiert, der nur permutative Reduktionen enthält, dann der volle Kalkül $\alpha \gg_n$ mit echten Reduktionen. Das Church-Rosser Theorem teilt sich dadurch in drei natürliche Teile:

Wie vertauschen zwei permutative Reduktionen ?

Wie vertauscht eine permutative mit einer echten Reduktion ?

Wie vertauschen zwei echte Reduktionen ?

Das Hauptproblem bei den permutativen Reduktionen wird in LEMMA 5 (Fall $j=3, k=4', l=3$) behandelt. Hier wird klar, warum als Maß für die Reduktionslänge in Reduktionsregel 2) die Norm des Hauptteiles auftritt.

Die spezielle Definition der Norm ist hierauf abgestimmt.
 Der Preis für diese Definition wird im selben LEMMA 5 (Fall $j=3$,
 $k=4'$, $l=4'$) sichtbar: Die Norm muß so definiert sein, daß sie
 bei permutativen Reduktionen nicht wächst (LEMMA 1).
 Damit sind die folgenden Ausführungen unmittelbar verständlich.

1. ---Reduktion $\alpha \vdash_n$ -

Der Reduktionskalkül $\alpha \vdash_n$ behandelt die permutativen Konversionen.
 "Echte Konversionen sind in ihm noch nicht zugelassen. Die Länge
 von Reduktionsfolgen wird durch zwei untere Indices $\alpha \vdash_n$ gemessen.

DEFINITION $\alpha \vdash_n$ (α Ordinalzahl, $n \in \mathbb{N}$, $n \geq 1$)

Bem: Hier und im folgenden sei $\alpha \vdash$ ein Abkürzung für $\alpha \vdash_1$.

1) $r \alpha \vdash r$ für alle α (Ordinalzahl) und Terme r .

3) $B(B'(t, (\vec{y}_1)_{s_1}, (\vec{y}_2)_{s_2}, \dots), (\vec{z}_1)_{u_1}, (\vec{z}_2)_{u_2}, \dots) \parallel t \parallel \vdash$
 $B'(t, (\vec{y}_1)_{B(s_1, (\vec{z}_1)_{u_1}, (\vec{z}_2)_{u_2}, \dots)}, (\vec{y}_2)_{B(s_2, (\vec{z}_1)_{u_1}, (\vec{z}_2)_{u_2}, \dots)}, \dots)$
 für jede Beseitigungskonstante B und jede kritische
 Beseitigungskonstante B' .

4) $t_1 \alpha \vdash t'_1, t_2 \alpha \vdash t'_2, \dots \Rightarrow$
 $E((\vec{x}_1)_{t_1}, (\vec{x}_2)_{t_2}, \dots) \alpha \vdash E((\vec{x}_1)_{t'_1}, (\vec{x}_2)_{t'_2}, \dots)$
 für jede Einführungskonstante E .

4') $t \alpha \vdash t' \Rightarrow$
 $B(t, (\vec{y}_1)_{s_1}, (\vec{y}_2)_{s_2}, \dots) \alpha \vdash B(t', (\vec{y}_1)_{s_1}, (\vec{y}_2)_{s_2}, \dots)$
 für jede Beseitigungskonstante B .

4'') $s_1 \alpha \vdash s'_1, s_2 \alpha \vdash s'_2, \dots \Rightarrow$
 $B(t, (\vec{y}_1)_{s_1}, (\vec{y}_2)_{s_2}, \dots) \alpha \vdash B(t, (\vec{y}_1)_{s'_1}, (\vec{y}_2)_{s'_2}, \dots)$
 für jede Beseitigungskonstante B .

5) $r \alpha \vdash s$ nach einer der Regeln 1 bis 4"
 und $s \beta \vdash_n r'$ mit $\beta < \alpha \Rightarrow r \alpha \vdash r'$.

5') $r \alpha \vdash s$ und $s \alpha \vdash_n r' \Rightarrow r \alpha \vdash_{n+1} r'$.

Im folgenden steht $r \alpha \vdash_n^i r'$ abkürzend für: $r \alpha \vdash_n r'$ wurde zuletzt mit Regel i erschlossen.

LEMMA 1. "Reduktion und Norm"

$$r \alpha \vdash_n r' \Rightarrow \|r\| \geq \|r'\|.$$

Beweis:

Induktion über $\omega\alpha + n$, Nebeninduktion über den Aufbau des Termes r ,
 Fallunterscheidung nach der Regel i mit $r \alpha \vdash_n^i r'$.

i=1: trivial

i=3:

$$\begin{aligned} & \|B(B'(t, (\vec{y}_1)_{s_1}, \dots), (\vec{z}_1)_{u_1}, \dots)\| = \\ & \sup\{\|u_j\|\} + (\sup\{\|s_1\|\} + \|t\| + 1) + 1 \geq \\ & \sup\{\sup\{\|u_j\|\} + \|s_1\| + 1\} + \|t\| + 1 = \\ & \|B'(t, (\vec{y}_1)_{B(s_1, (\vec{z}_1)_{u_1}, \dots), \dots})\|. \end{aligned}$$

i=4, 4', 4'': mit Nebeninduktionsvoraussetzung.

i=5:

$r \alpha \vdash s$ nach einer der Regeln 1 bis 4" $\Rightarrow \|r\| \geq \|s\|$ nach den schon behandelten Fällen i=1, 3, 4, 4', 4''.

$s \beta \vdash_n r'$ mit $\beta < \alpha \Rightarrow \|s\| \geq \|r'\|$ nach Induktionsvoraussetzung.

Damit folgt $\|r\| \geq \|r'\|$.

i=5': mit Induktionsvoraussetzung.

LEMMA 2. "Verlängerung von Reduktionsketten"

$$r \beta \vdash_m r' \Rightarrow r \alpha \vdash r' \text{ für } \alpha > \beta.$$

Beweis:

$$r \overset{1}{\alpha} \vdash r \beta \vdash_m r' \Rightarrow r \overset{5}{\alpha} \vdash r'.$$

LEMMA 3. "Transitivität und Länge"

$$r \alpha \vdash s \beta \vdash_m r' \text{ mit } \beta < \alpha \Rightarrow r \alpha \vdash r'.$$

Beweis:

Ist $r \overset{i}{\alpha} \vdash s$ mit $i=1,3,4,4',4''$, so folgt die Behauptung mit Regel 5.
 $i=5'$ scheidet aus da sonst $r \alpha \vdash_{n+1} s$ und $n+1 > 1$.

Bleibt $i=5$:

Also $r \overset{k}{\alpha} \vdash s' \delta \vdash_n s$ mit $\delta < \alpha$, $k=1, \dots, 4$

Es folgt $s' \delta \vdash_1 r'$ mit $\delta = \max\{\delta, \beta\}$, $1 \in \mathbb{N}$ mit LEMMA 2;

somit $r \overset{k}{\alpha} \vdash s' \delta \vdash_1 r'$ mit $\delta < \alpha$ also $r \overset{5}{\alpha} \vdash r'$.

LEMMA 4. "Inversion"

$$K((\vec{x}_1)t_1, (\vec{x}_2)t_2, \dots) \alpha \vdash_n K((\vec{x}_1)t'_1, (\vec{x}_2)t'_2, \dots) \Rightarrow t_i \alpha \vdash_n t'_i \text{ für } i = 1, 2, \dots \text{ falls } K \text{ eine Einführung oder eine nicht kritische Beseitigung ist.}$$

Beweis:

Induktion nach $\omega\alpha + n$,

Fallunterscheidung nach der zuletzt benutzten Regel k .

$k=1,4,4',4''$: Die Behauptung ist Voraussetzung für die Regelanwendung.

$k=3$: Da K keine kritische Beseitigung ist, in der Regel 3 aber rechts von $\vdash$ stets eine kritische Beseitigung steht, kann der Fall $k=3$ nicht auftreten.

$k=5$: Weil K keine kritische Beseitigung ist, folgt aus

$s \beta \vdash_m (\dots)$, daß auch s die Gestalt $K(\dots)$ hat. Die Behauptung folgt dann sofort (die Regeln 1 bis 4" sind schon behandelt und wegen $\alpha > \beta$ ist die Induktionsvoraussetzung anwendbar) mit Lemma 3.

$k=5'$: Folgt direkt aus der Induktionsvoraussetzung und Regel 5'.

Um die Beweise im folgenden übersichtlicher zu machen, gehe ich davon aus, daß die Einführungsformen die Wertigkeit (1) und die Beseitigungsformen die Wertigkeit (0,1) besitzen.

Der Unterschied zum allgemeinen Kalkül besteht lediglich darin, daß man statt $E((x_1^1, \dots) t_1, \dots)$ einfach $E((x)t)$ und statt $B(t_0, (x_1^1, \dots) t_1, \dots)$ einfach $B(t_0, (x)t_1)$ schreibt. Man spart sich also das Hantieren mit Indices, Pfeilen und "...".

LEMMA 5. "Church-Rosser Eigenschaft für $\vdash$ "

$$r \propto_n r' \text{ und } r \beta_m r'' \Rightarrow$$

Es gibt ein s mit $r' \beta_m s$ und $r'' \propto_n s$.

Beweis:

Induktion nach $\alpha \# \beta$.

1 Fall $n = m = 1$.

Nebeninduktion nach $\|r\|$:

Fallunterscheidung nach j und k mit $r \propto^j r'$ und $r \beta^k r''$.

Es werden im folgenden nur einige typische Fälle ausgeführt.

$j=3, k=4'$:

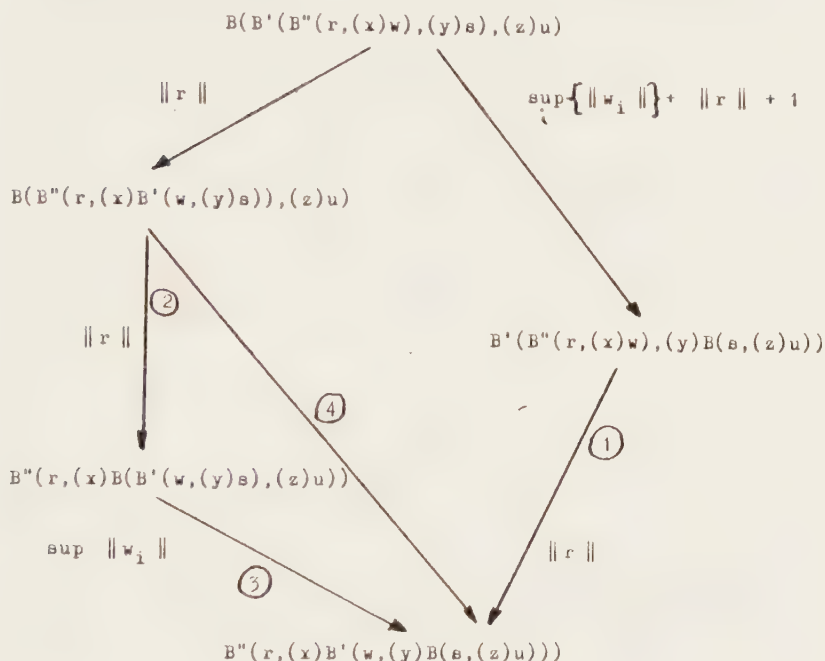
$$r = B(B'(t, (y)s), (z)u) \parallel_t \beta^3 B'(t, (y)B(s, (z)u))$$

und

$$B(B'(t, (y)s), (z)u) \beta^{4'} B(v, (z)u) \\ \text{mit } B'(t, (y)s) \beta^1 v.$$

Fallunterscheidung nach 1:

1=3: Der Beweis kann folgendem Diagramm entnommen werden:



Dabei folgt (1) und (2) mit Regel 3).

(3) folgt mit Regel 4") wobei die Prämissen dieser Regel mit Regel 3) und Lemma 2 folgt. Die Behauptung (4) folgt dann mit Lemma 2. (2) gilt auch mit Länge $\sup \{ ||w_i|| \} + ||r|| + 1$ wegen Lemma 3.)

1=4':

$B'(t, (y)s) \stackrel{4'}{\beta} B'(t', (y)s)$

impliziert $t \beta t'$.

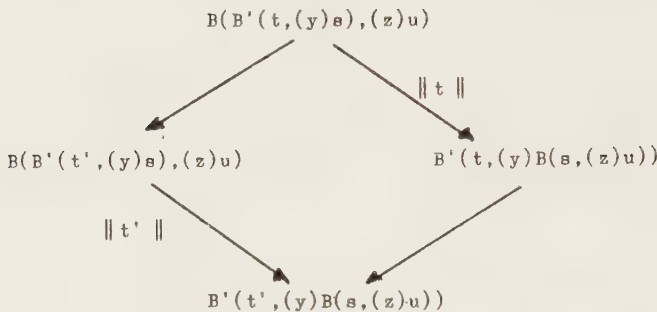
Damit $B'(t, (y)B(s, (z)u)) \stackrel{4'}{\beta} B'(t', (y)B(s, (z)u))$.

Aus

$B(B'(t', (y)s), (z)u) \stackrel{3}{\beta} B'(t', (y)B(s, (z)u))$

und $||t|| \geq ||t'||$ (Lemma 1) folgt mit Lemma 2 schließlich

der Rest der Behauptung.



1=5:

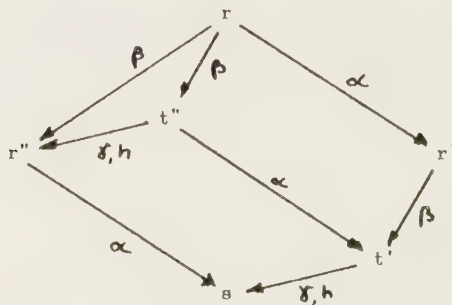
Also $B'(t, (y)s) \stackrel{i}{\beta} v' \delta \vdash_h v$ mit $i=1, \dots, 4$, $\delta < \beta$.

Mit $t'' = B(v', (z)u)$ gilt $r \stackrel{4}{\beta} t'' \stackrel{4}{\delta} \vdash_h r''$.

Nachdem die Fälle $1=1, \dots, 4$ behandelt sind, folgt die Existenz eines t' mit $r' \beta \vdash t'$ und $t'' \alpha \vdash t'$.

Die Existenz eines s mit $r'' \alpha \vdash s$ und $t' \delta \vdash_h s$ folgt dann unter Benutzung der Induktionsvoraussetzung ($\delta < \beta$) mit einer einfachen Induktion nach h . $r' \beta \vdash s$ schließlich folgt aus $r' \beta \vdash t' \delta \vdash_h s$ mit Lemma 3.

Diagramm:



j=4, k=4 :

$E((x)t) \stackrel{4}{\alpha} \vdash E((x)t')$ mit $t \alpha \vdash t'$ und
 $E((x)t) \stackrel{4}{\beta} \vdash E((x)t'')$ mit $t \beta \vdash t''$.

Hier folgt die Behauptung sofort mit LEMMA 4 und der Nebeninduktionsvoraussetzung, gefolgt von einer Anwendung der Regel 4.

Die übrigen Fälle mit $j, k < 5'$ werden ganz analog behandelt.

2 Fall $n = 1, m > 1$.

Beweis durch Induktion nach m . Wegen $m > 1$ kommt für $r \beta_m r''$ nur die Regel 5' in Frage. Man hat somit $r \beta t'' \beta_{m-1} r''$. Nach Fall 1 gibt es ein s'' mit $r' \beta s''$ und $t'' \alpha s''$ und nach Induktionsvoraussetzung bezüglich m gibt es ein s mit $s'' \beta_{m-1} s$ und $r'' \alpha s$. Aus $r' \beta s'' \beta_{m-1} s$ folgt $r' \beta_m s$.

3 Fall $n > 1, m > 1$

Beweis analog zu Fall 2 mit Induktion nach n .

Als Vorbereitung für die echten Reduktionen muß man sich noch die Substitution genauer anschauen.

Die spezielle Definition der Reduktionslänge in Regel 3 durch die Norm des Hauptteiles verhindert beim folgenden Lemma Aussagen über die Reduktionslängen nach der Substitution. Beim Übergang zum vollen Reduktionskalkül » spielt dies jedoch keine Rolle mehr (siehe unten Regel 3).

LEMMA 6. "Substitution"

$r \alpha_n r'$ und $s \beta_m s' \Rightarrow$ Es gibt ein γ mit $r_X[s] \gamma r'_X[s']$.

Beweis:

Man beachte im folgenden Lemma 2. Seine Verwendung ist nicht immer explizit gemacht.

Wegen $r_X[s] \beta r_X[s'] \beta r'_X[s'] \Rightarrow r_X[s] \beta r'_X[s']$ kann die Behauptung in zwei Teilen bewiesen werden.

Teil 1: $r_X[s] \beta r_X[s']$ durch Induktion nach $\|r\|$.

1) $r = x \Rightarrow r_X[s] = s \beta_m s' = r_X[s']$.

2) $r = y \dagger x \Rightarrow r_X[s] = y \gamma y = r_X[s']$.

3) $r = K((\vec{x}_1)t_1, (\vec{x}_2)t_2, \dots)$. Nach Induktionsvor. gilt

$t_{iX}[s] \beta t_{iX}[s']$ also mit $\gamma = \sup\{\delta_i\}$ folgt mit Regel 4, 4', 4"

$$r_x[s] = K((\vec{x}_1)t_{1x}[s], (\vec{x}_2)t_{2x}[s], \dots) \stackrel{r}{\vdash} K((\vec{x}_1)t_{1x}[s'], (\vec{x}_2)t_{2x}[s'], \dots) \\ = r_x[s'].$$

Anmerkung: Die Aussage des Lemmas läßt sich dahingehend verschärfen, daß für Teil 1 der Behauptung $\gamma = \beta + 1$ genügt.

Teil 2 : $r_x[s] \vdash r'_x[s]$ durch Induktion nach $\omega\alpha + n$
Nebeninduktion nach $\|r\|$.

Fallunterscheidung nach i mit $r \stackrel{i}{\vdash}_n r'$.

$i=1$: trivial.

$i=3$:

$$r_x[s] = B(B'(t_x[s], (y)w_x[s]), (z)u_x[s]) \stackrel{3}{\vdash} \\ B'(t_x[s], (y)B(w_x[s], (z)u_x[s])) = r'_x[s].$$

wobei o.B.d.A. x von allen y und z verschieden.

$i=4, 4', 4''$: durch Anwenden der Nebeninduktionsvor. .

$i=5$:

$$r \stackrel{1}{\vdash}_\alpha v \stackrel{\delta}{\vdash}_h r' \quad \text{mit } \delta < \alpha \text{ und } l = 1, \dots, 4'' \Rightarrow$$

$$r_x[s] \stackrel{\alpha}{\vdash} v_x[s] \text{ nach den schon behandelten Fällen.}$$

$$v_x[s] \stackrel{\delta}{\vdash}_h r'_x[s] \text{ nach Induktionsvor.}$$

damit folgt die Behauptung (LEMMA 3) .

$i=5'$:

mit Induktionsvoraussetzung.

Anmerkung: Auch Teil 2 läßt sich ähnlich wie Teil 1 verschärfen:

Es genügt, $\gamma = \|r_x[s]\| + \alpha$ zu wählen.

Für das Lemma insgesamt genügt so das Maximum dieser Werte plus Eins.

2. Reduktion $\alpha \gg_n$

DEFINITION $\alpha \gg_n$ (α Ordinalzahl, $n \in \mathbb{N}$)

Bem: Hier und im folgenden sei $\alpha \gg$ eine Abkürzung für $\alpha \gg_1$.

1) $r \alpha \gg r$ für alle $\alpha \geq 1$ und Terme r .

2) Sei H, H' eine Kontraktionsvorschrift des Kalküls, die die Metavariablen $Z_1, Z_2, \dots$ enthält, t_i, t'_i Terme mit $t_i \alpha \gg t'_i$ und $\vec{x}_i$ Listen von Variablen $i = 1, 2, \dots$, so gilt:
 $H_{Z_1, Z_2, \dots} [(\vec{x}_1) t_1, (\vec{x}_2) t_2, \dots] \alpha \gg H'_{Z_1, Z_2, \dots} [(\vec{x}_1) t'_1, (\vec{x}_2) t'_2, \dots]$

3) $r \alpha \vdash_n r' \Rightarrow r \alpha \gg r'$

4) $t_0 \alpha \gg t'_0, t_1 \alpha \gg t'_1, \dots \Rightarrow$
 $K((\vec{x}_0) t_0, (\vec{x}_1) t_1, \dots) \alpha \gg K((\vec{x}_0) t'_0, (\vec{x}_1) t'_1, \dots)$
 für jede Form K .

5) $r \alpha^1 \gg s$ mit $i = 1, 2, 3, 4$ und $s \beta \gg_n r'$ mit $\beta < \alpha \Rightarrow r \alpha \gg r'$

5') $r \alpha \gg s \alpha \gg_n r' \Rightarrow r \alpha \gg_{n+1} r'$

Im folgenden steht $r \alpha^1 \gg_n r'$ abkürzend für: $r \alpha \gg_n r'$ wurde zuletzt mit Regel i erschlossen.

Die Beweise der folgenden drei Lemmata sind analog zu den Beweisen der Lemmata 2 bis 4 und werden deshalb hier ausgelassen.

LEMMA 7. "Verlängerung von Reduktionsketten"

$$r \beta \gg_n r' \Rightarrow r \alpha \gg r' \quad \text{für } \alpha > \beta.$$

LEMMA 8. "Transitivität und Länge"

$$r \alpha \gg s \beta \gg_n r' \Rightarrow r \alpha \gg r' \quad \text{für } \alpha > \beta.$$

LEMMA 9 "Inversion"

$$E((\vec{x}_1)t_1, (\vec{x}_2)t_2, \dots) \alpha \gg E((\vec{x}_1)t'_1, (\vec{x}_2)t'_2) \Rightarrow \\ t_i \alpha \gg t'_i \quad \text{für } i=1, 2, \dots \text{ und alle Einführungsformen } E.$$

LEMMA 10 "Substitution"

$$t \alpha \gg_n t' \quad \text{und} \quad s \alpha \gg s' \Rightarrow t_X[s] \alpha \gg_n t'_X[s'].$$

Beweis:

Induktion nach $\omega\alpha + n$.

Nebeninduktion nach $\|t\|$.

Fallunterscheidung nach der zuletzt angewandten Regel i mit $t \overset{i}{\alpha} \gg_n t'$

$i=1$: $t = x \Rightarrow$ Behauptung ist gerade $s \alpha \gg s'$.

$t = y \dagger x \Rightarrow$ Behauptung ist gerade $t \overset{1}{\alpha} \gg t'$

$t = K(\dots, (x_j)t_j, \dots)$ mit Nebeninduktionsvoraussetzung und Regel 4.

$i=2$: mit Nebeninduktionsvoraussetzung.

$i=3$: $\alpha = 0 \Rightarrow$ Behauptung ist gerade Lemma 6.

$i=4$: mit Nebeninduktionsvoraussetzung.

$i=5$: $t \alpha \gg r$ nach Regel 1, 2, 3 oder 4 und $r \beta \gg_m t'$ mit $\beta < \alpha$.

$t_X[s] \alpha \gg r_X[s']$ nach den schon behandelten Fällen.

$r_X[s'] \beta \gg_m t'_X[s']$ mit Ind. Vor. (wegen $s' \overset{1}{\beta} \gg s'$)

Die Behauptung folgt nun mit Lemma 8.

$i=5'$: mit Induktionsvoraussetzung.

3. Church-Rosser Theorem

Nachdem bisher $\vdash$ und $\gg$ getrennt behandelt wurden, wird im folgenden Lemma die Vertauschbarkeit von echten und permutativen Reduktionen betrachtet.

Hier ist die Stelle, an der entscheidend die Definition der kritischen Regel eingeht. (Die Definition wurde gerade auf dieses Lemma zugeschnitten.)

LEMMA 11 "Mischung von $\vdash$ und $\gg$ "

$$r \alpha \vdash_n r' \text{ und } r \beta \gg_m r'' \Rightarrow$$

Es gibt ein s mit $r' \beta \gg_m s$ und $r'' \alpha \gg s$.

Beweis:

Induktion nach $\omega\beta + m$

Nebeninduktion nach $\|r\|$

Nebeninduktion nach $\omega\alpha + n$

Fallunterscheidung nach j mit $r \beta \gg_m^j r''$.

Bei dieser und den folgenden Fallunterscheidungen werden stets nur einige repräsentative Fälle behandelt.

$j=1$: trivial.

$j=3$: Behauptung folgt aus Lemma 5.

$j=2$:

Fallunterscheidung nach k mit $r \alpha \vdash_n^k r_1$.

$k=1, 4$: trivial.

$k=3, 4$: widerspricht $j=2$.

$k=4'$:

$$r = B(E((\vec{x}_1)t_1, \dots), (\vec{y}_1)s_1, \dots) \beta \gg H_{T_1}[(\vec{x}_1)t_1'] \dots S_1[(\vec{y}_1)s_1'] \dots$$

$$\text{mit } t_i \beta \gg t_i' \text{ und } s_i \beta \gg s_i'.$$

und

$$r = B(E((\vec{x}_1)t_1, \dots), (\vec{y}_1)s_1, \dots) \alpha \vdash^{4'} B(v, (\vec{y}_1)s_1 \dots)$$

mit $E((\vec{x}_1)t_1, \dots) \alpha \vdash v = E((\vec{x}_1)t_1'', \dots)$ da der Kalkül $\vdash$ Einführungsformen erhält.

Nach Nebenind.vor. und Lemma 4 gibt es nun t_i'' mit $t_i' \beta \gg t_i''$

und $t_i' \alpha \vdash t_i''$. Damit folgt: (mit Lemma 6)

$$r'' = H'_{T_1}[(\vec{x}_1)t_1'] \dots S_1[(\vec{y}_1)s_1'] \dots \vdash H'_{T_1}[(x_1)t_1''] \dots S_1[(y_1)s_1'] \dots$$

und

$$r' = H \ T_1[(\vec{x}_1)t_1'] \dots S_1[(\vec{y}_1)s_1] \dots \quad \gg \quad H' \ T_1[(\vec{x}_1)t_1''] \dots S_1[(\vec{y}_1)s_1'] \dots$$

mit Regel 2 .

k=5 :

unter Nutzung der Tatsache, daß die Fälle k=1...4" schon behandelt sind mit der Induktionsvoraussetzung.

k=5': mit Induktionsvoraussetzung.

j=4 :

Also :

$$r = K((\vec{x}_1)u_1, \dots) \beta \gg K((\vec{x}_1)u_1', \dots) = r''$$

mit $u_i \beta \gg u_i'$ für $i = 1, 2, \dots$.

Fallunterscheidung nach k mit $r \alpha_{\beta, n}^k r_1$.

k=1 : trivial.

k=3 :

Fallunterscheidung nach l mit $u_1 \beta^l \gg u_1'$.

l=1' :

mit Lemma 5.

l=2 :

Bei diesem Fall geht die Tatsache ein, daß die permutativen Reduktionen nur für kritische Beseitigungen zulässig sind.

Es ist

$$\begin{aligned} r &= B(B'(E((\vec{y}_1)t_1, \dots), (\vec{z}_1)s_1, \dots), (\vec{x}_2)u_2, \dots) = \\ &B(H \ T_1[(\vec{y}_1)t_1] \dots S_1[(\vec{z}_1)s_1] \dots, (\vec{x}_2)u_2, \dots) \beta \gg \\ &B(H' \ T_1[(\vec{y}_1)t_1'] \dots S_1[(\vec{z}_1)s_1'] \dots, (\vec{x}_2)u_2', \dots) = r'' \end{aligned}$$

wobei $u_i \beta \gg u_i'$, $s_i \beta \gg s_i'$ und $t_i \beta \gg t_i'$.

Daneben

$$\begin{aligned} r \alpha_{\beta, n} B'(E((\vec{y}_1)t_1, \dots), (\vec{z}_1)B(s_1, (\vec{x}_2)u_2, \dots), \dots) &= r' = \\ H \ T_1[(\vec{y}_1)t_1] \dots S_1[(\vec{z}_1)B(s_1, (\vec{x}_2)u_2, \dots)] \dots \end{aligned}$$

Sofort folgt

$$r' \beta^2 \gg H' \ T_1[(\vec{y}_1)t_1'] \dots S_1[(\vec{z}_1)B(s_1', (\vec{x}_2)u_2', \dots)] \dots = s .$$

Nachdem nun B' eine kritische Beseitigung ist, gibt es für H' zwei Möglichkeiten:

1) $H' = B'(R, (\vec{z}_1)(S_{i_1}(\vec{z}_1)), \dots)$ wo R nur die T_i enthält,
dann ist

$$r'' = B(B'(R_{T_1}[(\vec{y}_1)t_1] \dots, (\vec{z}_1)s_{i_1}', \dots), (\vec{x}_2)u_2', \dots) \quad \text{O} \gg$$

$$B'(R_{T_1}[(\vec{y}_1)t_1] \dots, (\vec{z}_1)B(s_{i_1}', (\vec{x}_2)u_2', \dots), \dots) =$$

$$H'_{T_1}[(\vec{y}_1)t_1] \dots S[(\vec{z}_1)B(s_{i_1}', (\vec{x}_2)u_2', \dots)] \dots = s.$$

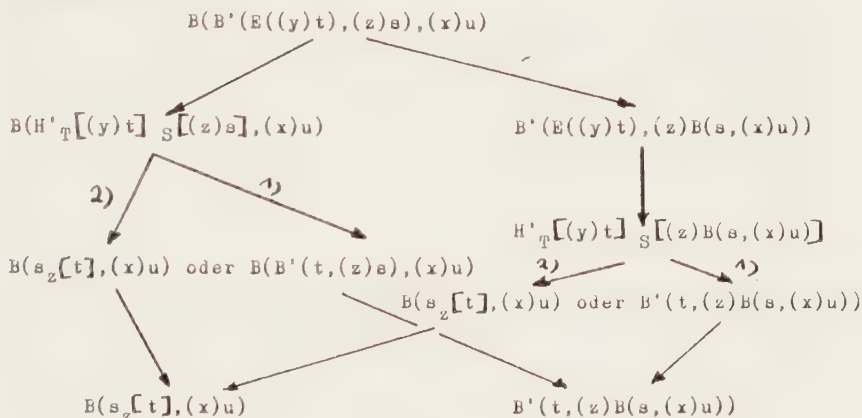
2) $H' = S_{i_0}(R_1, \dots)$ wo die R_i nur die T_i enthalten.

dann ist

$$r'' = B(S_{i_0} z [R_1 T_1[(\vec{y}_1)t_1] \dots] \dots, (\vec{x}_2)u_2', \dots) =$$

$$H'_{T_1}[(\vec{y}_1)t_1] \dots S[(\vec{z}_1)B(s_{i_1}', (\vec{x}_2)u_2', \dots)] \dots = s.$$

Diagramm:



$k=4, 4', 4''$: Mit Nebeninduktionsvor. bezüglich $\|r\|$.

$k=5, 5'$: Mit Induktionsvor. bezüglich $\omega\alpha + n$.

$j=5$: mittels der schon behandelten Fälle $j=1, 1', 2, 4$ und der Induktionsvor. bezüglich β .

$j=5'$: mittels Induktionsvor. bezüglich β .

SATZ "CHURCH-ROSSER THEOREM"

$r \alpha \gg_n r'$ und $r \beta \gg_m r'' \Rightarrow$

Es gibt ein s mit $r' \beta \gg_m s$ und $r'' \alpha \gg_n s$.

Beweis:

Induktion nach $\alpha \# \beta$.

$\alpha = 0$ oder $\beta = 0$: Behauptung folgt aus Lemma 11.

Sei also $\alpha \neq 0$ und $\beta \neq 0$.

$n=m=1$: Nebeninduktion nach $\|r\|$.

Fallunterscheidung nach k, j mit $r \alpha \gg^k r'$ und $r \beta \gg^j r''$.

$k=1$ oder $j=1$: trivial.

$k=2, j=2$:

Da eine Kontraktionsvorschrift durch die beiden zugehörigen Formen im Term eindeutig festgelegt ist, folgt die Behauptung direkt.

$k=2, j=4$:

$r = B(E((\vec{x}_1)t_1, \dots), (\vec{y}_1)s_1, \dots) \beta \gg B(v, (\vec{y}_1)s_1', \dots) = r''$

mit $E((\vec{x}_1)t_1, \dots) \beta \gg v$ und $s_1 \beta \gg s_1''$,

sowie

$r = H_{T_1}[(\vec{x}_1)t_1] \dots S_1[(\vec{y}_1)s_1] \dots \alpha \gg H'_{T_1}[(\vec{x}_1)t_1'] \dots S_1[(\vec{y}_1)s_1'] = r'$

Zuerst folgt, daß $v = E((\vec{x}_1)t_1'', \dots)$, da alle Regeln $\gg$ einen Term der Form $E(\dots)$ erhalten. Mit Lemma 9 folgt dann noch $t_i \beta \gg t_i''$.

Nun wendet man auf die t_i' und t_i'' sowie die s_i' und s_i'' die Ind.Vor. an und erhält so Terme t_i''' sowie Terme s_i''' .

Mit $s = H'_{T_1}[(x_1)t_1'''] \dots S_1[(y_1)s_1'''] \dots$

folgt die Behauptung dann zum einen mit Lemma 10 und zum anderen mit Regel 2.

$k \leq 5, j=5$:

Durch Rückgriff auf die schon behandelten Fälle $k, j=1, 2, 4$ und die Induktionsvor. erhält man direkt die Behauptung.

$k=5, j=5$:

Analog zum vorigen Fall unter Einschluß von $j=5$.

$m=1, n \geq 1$: Durch Induktion nach m .

$m \geq 1, n \geq 1$: Durch Induktion nach n .

Zur Übertragbarkeit des Church-Rosser Theorems auf modifizierte Kalküle gibt das folgende einfache Korollar Auskunft:

KOROLLAR

Sei $\mathcal{L}$ eine Menge von Objekten mit einer transitiven Relation $>$. Weiter sei τ eine Abbildung von $\mathcal{L}$ in die Terme des (eines) Termkalküls so, daß

a) $B > B' \Rightarrow \tau(B) \gg \tau(B')$ für alle B, B' aus $\mathcal{L}$

b) $\tau(B) \gg b' \Rightarrow \exists! B' \in \mathcal{L}: B > B' \wedge \tau(B') = b'$ für alle B aus $\mathcal{L}$

Dann gilt das Church-Rosser Theorem auch für $(\mathcal{L}, >)$.

Beweis:

Es gelte $B > B'$ und $B > B'' \Rightarrow \tau(B) \gg \tau(B')$ und $\tau(B) \gg \tau(B'')$.

Es gibt dann ein t mit $\tau(B') \gg t$ und $\tau(B'') \gg t$.

Dann gibt es eindeutig bestimmte T' und T'' aus $\mathcal{L}$ mit $B' > T'$ und $B'' > T''$ und $\tau(T') = t = \tau(T'')$.

Nun gilt aber auch $B > T'$ und $B > T''$ sowie $\tau(B) \gg t$. Wegen der Eindeutigkeit folgt so $T' = T''$.

FOLGERUNGEN

Das Church-Rosser Theorem gilt für den getypten Termkalkül und den Kalkül natürlicher Herleitungen, wie in den Beispielen von Kap. I.3 beschrieben.

Die Abbildung τ , die für eine Anwendung des Korollars gebraucht wird, ist trivial, ebenso Eigenschaft a).

Eigenschaft b) folgt für den getypten Termkalkül sofort daraus, daß B' durch b' und die Typen der freien Variablen eindeutig bestimmt ist. Die freien Variablen von b' sind aber sämtliche auch frei in B . Somit ist ihr Typ eindeutig bestimmt.

Für die natürlichen Herleitungen folgt b) mit einer analogen Überlegung, was die freien Variablen anbelangt, und der Feststellung, daß die Konklusion durch die Prämissen und die verwendete Schlussregel eindeutig bestimmt wird.

III. Normalisierung.

Den Anfang dieses Kapitels bildet ein Existenzsatz für Normalformen. Interessiert man sich weitergehend dafür, wie man solch eine Normalform erhalten kann, kommt man zu den Fragen nach starker Normalisierung (jede Reduktion führt zu einem normalen Term) und der stetigen Normalisierung.

In unserem Zusammenhang, d.h. Betonung auf unendlichen Termen, ist ein starker Normalisierungssatz von vorneherein problematisch.

In Teil 2 werden für den Termkalkül T_{∞} diese Probleme (und Ergebnisse) dargelegt.

Natürgemäß interessanter sind hier die stetigen Normalisierungen, die in Teil 3 behandelt werden. Der reine Existenzsatz von Teil 1 wird durch den stärkeren Satz in Teil 3 nicht uninteressant, denn um die Fundiertheit des normalen Termes zu beweisen, werden wieder die Methoden des ersten Teiles verwendet.

1. Existenz der Normalform

Der hier gegebene Existenzbeweis für Normalformen beruht auf der folgenden Idee, die auf eine Arbeit von Tait zurückgeht [9]:

Die Redexe werden geordnet nach ihrer Komplexität und man versucht dann, die größten Redexe so zu reduzieren, daß dabei höchstens kleinere Redexe entstehen (Lemma 1). Der Reduktionsrang erweist sich als das geeignete Maß für die Komplexität eines Redex.

Unter den Redexen gleicher Komplexität reduziert man von innen nach außen zuerst die kleinsten Teilterme.

Der eigentliche Normalformsatz kann dann durch eine Induktion nach der Norm eines Termes und seinem Reduktionsrang bewiesen werden. Für endliche Terme erhält man daraus sofort ein Normalisierungsverfahren ("maximal innermost"-Reduktion).

a) Definitionen

GETYPTER TERMKALKÜL

Die folgenden Ausführungen beziehen sich alle auf einen festen getypten Termkalkül wie in Kap. I beschrieben.

Die Zuordnung von Kontraktions schemata und Typsymbolen sei natürlich eineindeutig.

Um die Beweise übersichtlicher zu machen, gehe ich im folgenden wieder davon aus, daß die Einführungskonstanten die Stelligkeit (1) und die Beseitigungskonstanten die Stelligkeit (0,1) haben.

REDUKTION $\vdash$

- 1) $t \vdash t$ für alle t .
- 2) Ist H, H' eine Kontraktionsvorschrift des Kalküls und sind t, t' konsistente Spezialfälle von H, H' so $t \vdash t'$.
- 3) Sind B, B' Beseitigungen, wovon B' kritisch ist so gilt mit Termen r, s, t und Variablen x, y geeigneten Typs:

$$B(B'(r, (x)s), (y)t) \vdash B'(r, (x)B(s, (y)t))$$
- 4) Für jede Form K und Terme t_i :

$$t_0 \vdash t'_0, t_1 \vdash t'_1, \dots \Rightarrow$$

$$K((x_0)t_0, (x_1)t_1, \dots) \vdash K((x_0)t'_0, (x_1)t'_1, \dots).$$
- 5) $t \vdash s$ und $s \vdash t' \Rightarrow t \vdash t'$.

Bemerkung:

- 1) Es ist klar, daß ein getypter Term bei einer Reduktion wieder in einen getypten Term gleichen Typs übergeht. (Induktion über die Norm des Terms und die Länge der Reduktionskette.)
- 2) Der oben angegebene Reduktionskalkül ist zum Reduktionskalkül » aus Kap II äquivalent.

REDUKTIONSRANG

Ist ein Term $t = B(r, (y)s)$ nach einer der Regeln 2 oder 3
reduzierbar, so ist $n(r)$ der direkte Reduktionsrang $d(t)$ von t .
($n(t)$ ist das in Kap.I definierte Typniveau).

Ist t nicht nach einer dieser Regeln reduzierbar, so ist der direkte
Reduktionsrang von t gleich 0 .

Der Reduktionsrang $r(t)$ eines Terms t ist definiert als die
kleinste Zahl größer als die direkten Reduktionsränge aller Teil-
terme von t .

Im folgenden sind die Ränge von Termen als endlich vorausgesetzt.

NORMAL

Ein Term t heißt normal, wenn $r(t) = 1$.

(Er enthält dann keine echt reduzierbaren Teilterme mehr.)

NORMALISIERUNGSBEDINGUNG

Ein getypter Termkalkül erfüllt die Normalisierungsbedingung,
wenn:

- 1) Ist B eine beliebige Beseitigungskonstante, H, H' eine
Kontraktionsvorschrift für diese Konstante und enthält H'
einen Teilterm $T(H_1, H_2, \dots)$, so ist für beliebige
konsistente Spezialfälle $B(t_0, t_1, \dots), s_1, s_2, \dots$ von $H, H_1, H_2, \dots$
 $n(t_0)$ stets größer als $n(s_i)$ für jedes $i=1, 2, \dots$.

Die hier geforderte Beschränkung der Typen von eingesetzten
Termen ist nötig, damit die bei einer echten Reduktion neu
entstehenden Redexe "kleiner" werden (siehe Lemma 2).

- 2) Ist B eine nicht kritische Beseitigungskonstante mit Typschema
 $(\sigma_0, (\sigma_1^1, \dots) \sigma_1, \dots) \sigma$ so ist für beliebige konsistente
Spezialfälle μ_0 und μ von σ_0 und σ stets $n(\mu) \leq n(\mu_0)$.
D.h. Das Typniveau des Hauptteiles einer nicht kritischen Be-
seitigung ist größer oder gleich dem Typniveau des ganzen Termes.

Diese Bedingung ist speziell nötig zum Beweis von Lemma 3 .

In den folgenden Teilen, III.1 und III.2, wird von allen Termkalkülen
angenommen, daß diese beiden Bedingungen erfüllt sind.

b) Vorbereitende Lemmata

LEMMA 1

Seien s und t Terme, x eine Variable vom selben Typ wie s .
Ist dann $r(s) \leq k$, $r(t) \leq k$ und $n(s) < k$ so ist $r(t_x[s]) \leq k$.

Beweis:

Induktion über $\|t\|$, Fallunterscheidung nach dem Aufbau von t :

1) t Variable: trivial.

2) $t = E((z)v)$, E eine Einführung, $\Rightarrow t_x[s] = E((z)v_x[s]) \Rightarrow$
 $\Rightarrow r(t_x[s]) = r(v_x[s]) \leq k$ nach Induktionsvor.

3) $t = B(u, (z)v)$, B eine Beseitigung $\Rightarrow t_x[s] = B(u_x[s], (z)v_x[s])$

Sei nun r ein beliebiger Teilterm von $t_x[s]$, der nach Regel 2 oder 3
reduzierbar ist.

Ist r ein Teilterm von $u_x[s]$ oder von $v_x[s]$ so folgt $d(r) < k$
nach Induktionsvoraussetzung.

Ist $r = t_x[s]$ so ist

- entweder $u = K(\dots)$ mit K kritische Beseitigung/Einführung.

Damit ist $d(r) = d(t) < k$ nach Voraussetzung

- oder $u = x$ und also $u_x[s] = s$.

Dann ist $d(r) = n(u) = n(s) < k$.

Aus $d(r) < k$ für einen beliebigen Teilterm r von $t_x[s]$ folgt
aber $r(t_x[s]) \leq k$.

LEMMA 2

Sei $t = B(u, (z)v)$, B eine Beseitigung. Ist dann $r(u) < k$, $r(v) < k$
und $n(u) < k$, so existiert ein t' mit $t \triangleright t'$ und $r(t') < k$.

Beweis:

Zu betrachten ist nur der Fall $r(t) = k$.

Induktion nach $\|t\|$,

Fallunterscheidung nach der Gestalt von u :

1) $u = E((y)w)$ E eine Einführung.

Es gilt $t \triangleright t'$ nach Regel 2 für ein t' .

Mit LEMMA 1 der ersten Normalisierungsbedingung ($n(u) < k$)

und $r(u) < k$, $r(v) < k$ folgt sofort $r(t') < k$.

2) $u = B'(s, (y)w)$ B' kritische Beseitigung.

Dann ist t nach Regel 3 reduzierbar auf $B'(s, (y)B(w, (z)v))$.

Nach Induktionsvoraussetzung gilt $B(w, (z)v) \vdash w'$ mit $r(w') < k$.

Nun ist $r(s) \leq r(u) < k$ und $r(w') < k$. Mit $d(B'(s, (y)w')) \leq d(u) < r(u) < k$ folgt $r(t') = r(B'(s, (y)w')) < k$.

LEMMA 3

Sei $t = B(u, (z)v)$, B eine Beseitigung, und $n(u) \geq k$ und $r(t) \leq k$.

Dann hat u die Gestalt $u = B_1(B_2(\dots B_m(y, \dots) \dots))$ mit $m \geq 0$, B_1 nicht kritische Beseitigungen und y eine Variable.

Beweis:

Trivial gilt $u = B_1(B_2(\dots B_m(u', \dots) \dots))$ für ein $u' \vdash B_{m+1}(\dots)$

B_{m+1} nicht kritische Beseitigung (Man beachte die Möglichkeit $m=0$).

Aus der zweiten Normalisierungsbedingung folgt:

$$\begin{aligned} n(u) &= n(B_1(B_2(\dots B_m(u', \dots) \dots))) \leq n(B_2(\dots B_m(u', \dots) \dots)) \\ &\leq \dots \leq n(B_m(u', \dots)) \leq n(u') \end{aligned}$$

Ist nun $u' = E(\dots)$ mit einer Einführung E oder $B'(\dots)$ mit einer kritischen Beseitigung B' , so ist $B_m(u', \dots)$ nach Regel 2 bzw. 3 reduzierbar und es folgt $r(t) > d(B_m(u', \dots)) = n(u') \geq n(u) \geq k$ im Widerspruch zur Voraussetzung. Also ist u' eine Variable.

c) Existenzsatz

SATZ Existenz der Normalform

Sei t ein Term, dann gibt es einen normalen Term t' mit $t \vdash t'$.

Beweis:

Behauptung:

Sei t ein Term mit $r(t) \leq k$ so gibt es einen Term t' mit $t \vdash t'$ und $r(t') < k$.

Beweis durch Induktion nach k , eingeschachtelte Induktion nach $\|t\|$ und Fallunterscheidung nach dem Aufbau von t :

1) $t = x$ eine Variable:

trivial $t \vdash t = t'$.

2) $t = E((z)v)$, E eine Einführung.

Da $\|v\| < \|t\|$ folgt die Behauptung mit der Induktionsvoraussetzung

3) $t = B(u, (z)v)$, B eine Beseitigung .

Zuerst gibt es wegen $\|u\| < \|t\|$ und $\|v\| < \|t\|$ nach Induktionsvoraussetzung Terme u' und v' mit

$u \vdash u'$ und $v \vdash v'$ und $r(u') < k$ und $r(v') < k$.

Es folgt mit Regel 4') :

$t = B(u, (z)v) \vdash B(u', (z)v') = t''$.

Fall 1: $n(u') \geq k$

Nach LEMMA 3 folgt $u' = B_1(B_2(\dots B_m(y, \dots) \dots))$

Damit ist $r(B(u', (x)v')) = \max \{ r(u'), r(v') \} < k$ und die

Behauptung gilt mit $t' = t''$.

Fall 2: $n(u') < k$.

Man hat mit LEMMA 2 die Existenz von t' mit $t'' \vdash t'$

und $r(t') < k$. $t \vdash t'' \vdash t' \Rightarrow t \vdash t'$ wie behauptet.

Die Behauptung des Satzes folgt nun durch einfache Induktion nach $r(t)$ wegen $r(t)=1 \Rightarrow t \text{ normal}$.

Bemerkung:

Der hier gegebene Beweis wurde für endlichen Rang k geführt.

Streicht man die Einschränkung auf endlichen Rang in der Definition des Reduktionsranges und erweitert die Definition von Typen dahingehend, daß auch Typsymbole mit unendlicher Stelligkeit zugelassen werden (also unendliche Typniveaus möglich sind), so ist der dann nötige Beweis des Existenzsatzes von dem hier gegebenen nur geringfügig verschieden.

Eine Modifikation wird nur dort nötig, wo implizit - durch die Einschränkung auf endliche Terme aber nicht sichtbar - ein Schluß der folgenden Form verwendet wurde:

$$\begin{aligned} r(K(\dots, (\vec{z}_1)v_1, \dots)) \leq k &\Rightarrow r(v_1) \leq k \text{ für alle } i \Rightarrow \dots \Rightarrow \\ r(v'_1) < k \text{ für alle } i &\Rightarrow (!) \sup_i \{ r(v'_1) \} < k \Rightarrow \dots \Rightarrow \\ r(K(\dots, (\vec{z}_1)v'_1, \dots)) &< k . \end{aligned}$$

Der Übergang bei $(!)$ gilt natürlich nur für Nachfolgerzahlen k

Im Fall transfiniten Ränge setzt man $k = m + \omega^n$, mit Ordinalzahlen m und n .

Die Behauptung lautet nun: Ist $r(t) \leq m + \omega^n$, so gibt es t' mit $t \vdash t'$ und $r(t') \leq m$.

Dies wird durch Induktion nach n bewiesen:

$n=0$ ist gerade der oben behandelte Nachfolger-Fall.

$n>0$ wird völlig analog behandelt. Der oben erwähnte Schluß ist nun aber gültig, da $r(v'_i) \leq m$, m unabhängig von $i \Rightarrow$

$$\sup_i \{r(v'_i)\} \leq m.$$

Damit gilt der Existenzsatz auch für transfinite Ränge.

HÖHENABSCHÄTZUNG

Eine Höhenabschätzung für die normalen Terme ist eine Funktion $\mathcal{Y}$, so, daß im obigen Satz mitbewiesen werden kann $\|t'\| \leq \mathcal{Y}(\|t\|)$. (Hier ist das t' aus dem Induktionsbeweis gemeint mit $r(t') < k$. Für das normale t' ist die Abschätzung von $\|t'\|$ dann die $r(t)$ -fache Iteration von von obigem $\mathcal{Y}$.)

Für die Punkte 1,2 und Fall 1 von Punkt 3 eine Funktion $\mathcal{Y}$ mit $\mathcal{Y}(\|t\|) \geq \|t\|$.

Für Fall 2 von Punkt 3 muß man genauer das Lemma 2 betrachten. Dort sieht man die Abhängigkeit von den Kontraktionsvorschriften zum einen und von den permutativen Reduktionen zum anderen. Jede Funktion η , die den folgenden Ungleichungen genügt:

(mit den Bezeichnungen aus Lemma 2)

a) für die echten Reduktionen

$$\eta(\|t\|) \geq \mathcal{R}(\|t\|), \quad \mathcal{R} \text{ abhängig von den echten Reduktionen.}$$

b) für die permutativen Reduktionen

$$\eta\left(\sup_i \{\|v_i\|\} + \sup_j \{\|s_j\|\} + \|w\| + 1 + 1\right) \geq \sup_j \{\eta(\sup_i \{\|v_i\|\} + \|s_j\| + 1)\} + \|w\| + 1$$

ist eine mögliche Höhenabschätzung für Lemma 2 so, daß (mit den dortigen Bezeichnungen) gilt $\|t'\| \leq \eta(\|t\|)$. Dies kann man unmittelbar dem Beweis von Lemma 2 entnehmen.

Betrachtet man nun den Fall 2 von 3) im Existenzsatz, sieht man direkt, daß für die gesuchte Höhenabschätzung jede Funktion $\mathcal{Y}$ in Frage kommt, die der Ungleichung (Bezeichnungen wie im Satz)

$$\mathcal{Y}\left(\sup_i \{\|w_i\|\} + \|v\| + 1\right) \geq \eta\left(\sup_i \{\mathcal{Y}(\|w_i\|)\} + \mathcal{Y}(\|v\|) + 1\right)$$

genügt.

Ein Beispiel:

Für $\mathcal{R}$ sei gegeben $\mathcal{R}(\|t\|) = m\|t\|$ (etwa m -fach verschachtelte

Einsetzung)

Es ergibt sich daraus $\eta(\|t\|) = m \cdot \|t\|$ und $\mathcal{Y}(\|t\|) = m \|t\|$.

Die Induktion nach $r(t)$ bewirkt schließlich noch eine $r(t)$ -fache Iteration von $\mathcal{Y}$. Also:

$$\|t'\| \leq m \cdot \left. \begin{matrix} \|t\| \\ \vdots \\ \|t\| \end{matrix} \right\} r(t)$$

Man hat so eine $\mathcal{E}_4$ -Schranke für $\|t'\|$.

Bekannt ist weiter, daß eine $\mathcal{E}_4$ -Schranke schon bestmöglich ist, falls der Kalkül den λ -Kalkül umfaßt (siehe etwa [8]).

2. Starke Normalisierung

Unter einem starken Normalisierungssatz versteht man üblicherweise einen Satz der Form " Alle Reduktionsfolgen $t = t_0 \rightarrow t_1 \rightarrow t_2 \rightarrow \dots$ brechen nach endlich vielen Schritten mit einem normalen Term ab. " Betrachtet man nun auch unendliche Terme, so wird ein solcher Satz im allgemeinen nicht gelten - jedenfalls nicht bei den üblichen Definitionen von $\rightarrow$.

Ein Beispiel mag dies erläutern:

Man betrachte den getypten Termkalkül T_∞ , d.h. den λ -Kalkül erweitert um Nummerale $\underline{n}$.

Man bildet nun Terme

$$t_0 := 0$$

$$t_1 := (\lambda x_1. x_1) 0 \quad =: u_1 u_0$$

$$t_2 := (\lambda x_2. x_2 0) (\lambda x_1. x_1) \quad =: u_2 u_1$$

.

.

$$t_n := (\lambda x_n. x_n u_{n-2}) u_{n-1} \quad =: u_n u_{n-1}$$

Jedes t_i hat genau einen Redex und durch Kontraktion dieses Redex erhält man $t_i \rightarrow t_{i-1}$.

Der Term $\langle t_i \rangle y$ hat so die eindeutige Normalform $\langle t_0 \rangle y$ und jede Reduktionsfolge von $\langle t_i \rangle y$ enthält für alle i eine Reduktionsfolge von t_i . Hat $\langle t_i \rangle y$ eine endliche Reduktionsfolge der Länge n , so hat auch jedes t_i eine Reduktionsfolge der Länge n , was der intuitiven Vorstellung von einer Reduktionsfolge widerspricht. In ähnlicher Weise kann man aus beliebigen Termen s_i einen Term $\langle s_i' \rangle y$ aufbauen, so daß gilt:

Jede Reduktionsfolge von $\langle s_i' \rangle y$ enthält für alle s_i eine Reduktionsfolge. Sind also die Längen der kürzesten Reduktionsfolgen für die s_i nicht beschränkt, so ergibt dies einen Widerspruch zur Annahme $\langle s_i' \rangle y$ habe eine endliche Reduktionsfolge.

Unter den natürlichen Anforderungen an einen Reduktionsbegriff, daß

- man jeden Term auch komplizierter machen kann, so daß die Reduktionsfolge länger wird und
- eine Teiltermeigenschaft gilt wie : $t[s] \rightarrow t[s']$ in einem Schritt, so auch $s \rightarrow s'$ in einem Schritt,

kann also ein starker Normalisierungssatz nicht gelten.

Anders sieht die Sache aus, wenn man nicht alle Terme betrachtet, sondern sich auf geschlossene Terme vom Typ 0 beschränkt. Hier gibt es einen natürlichen Reduktionsbegriff, der im genannten Beispiel starke Normalisierung ermöglicht. Da der Termkalkül T_{∞} eine besondere Wichtigkeit hat, soll dies hier skizziert werden.

Die Haupt Schwierigkeit des folgenden Satzes liegt in der passenden Definition eines Reduktionsbegriffes.

Die Idee ist dabei, für Terme $\langle u_i \rangle t$ zuerst eine Abschätzung für den numerischen Wert von t zu machen. Terme u_i zu reduzieren, wenn der Index i größer ist als diese Abschätzung, wird nicht erlaubt. So beschränkt sich die Reduktion auf einen endlichen Teil des Terms; ein starker Normalisierungssatz wird möglich.

Vergleicht man den eingeschränkten Reduktionsbegriff mit dem üblichen Reduktionsbegriff, sieht man, daß jede normale Reduktionsfolge eine eingeschränkte Reduktionsfolge als Teilfolge enthält. So enthält jede beliebige Reduktionsfolge, anschaulich gesagt, nur endlich viele "sinnvolle" Schritte.

Für den Beweis selbst verwende ich eine Idee von R.O.Candy aus seiner Arbeit "Proofs of Strong Normalization" [2]. Es wurden Modifikationen vorgenommen, um die Beweismethode Gandys (Zuordnung von monotonen Funktionalen zu Termen) auf den Fall unendlicher Terme und permutative Reduktionen zu erweitern.

Leider gelingt es nicht, den Beweis auch für allgemeinere Kalküle durchzuführen, da für die Zuordnung der monotonen Funktionalen ein allgemeines Schema fehlt. Doch sind die im Reduktionsbegriff und in der Beweismethode sichtbaren Analysen der Termreduktion interessant genug, um eine Darstellung hier zu rechtfertigen.

Dem folgenden liegt der Termkalkül T_{∞} zugrunde.

MONOTONE FUNKTIONALE

Für jeden Typ σ wird F_{σ} definiert durch: $F_0 := \mathbb{N}$, und

$F_{\sigma \rightarrow \tau}$ ist die Menge der Funktoren von F_{σ} nach F_{τ} .

$F = \bigcup_{\sigma} F_{\sigma}$ ist abgeschlossen unter λ -Abstraktion und Anwendung.

$M \subset F$, die Menge der monotonen Funktoren wird zusammen mit einer partiellen Ordnung $<$ definiert durch:

$$1) M_0 = F_0 = \mathbb{N}, \quad <_0 = <_{\mathbb{N}}.$$

$$2) M_{\sigma \rightarrow \tau} = \{ \alpha \in F_{\sigma \rightarrow \tau} / \forall \gamma, \beta \in M_{\sigma} : \alpha \beta \in M_{\tau} \wedge (\gamma \leq_{\sigma} \beta \Rightarrow \alpha \gamma \leq_{\tau} \alpha \beta) \},$$

für $\alpha, \beta \in M_{\sigma \rightarrow \tau}$ gilt $\alpha <_{\sigma \rightarrow \tau} \beta$ genau dann wenn $\forall \gamma \in M_{\sigma} : \alpha \gamma <_{\tau} \beta \gamma$.

$$3) M = \bigcup_{\sigma} M_{\sigma}, \quad < = \bigcup_{\sigma} <_{\sigma}$$

Es ist klar, daß M abgeschlossen ist unter Anwendung und $<$ eine partielle Ordnung auf M definiert.

Ersetzt man in 1) $<_{\mathbb{N}}$ durch $\leq_{\mathbb{N}}$, so erhält man die Definition von einer partiellen Ordnung $\leq$ auf M .

SPEZIELLE MONOTONE FUNKTIONALE

$$1) +_{\sigma} \in M_{\sigma \rightarrow \sigma \rightarrow \sigma} \text{ definiert durch:}$$

$$+_0 = +_{\mathbb{N}}, \quad +_{\sigma \rightarrow \tau} = \lambda \alpha. \lambda \beta. \lambda \gamma. (\alpha(\gamma) +_{\tau} \beta(\gamma)).$$

$$2) S_{\sigma} \in M_{\sigma \rightarrow \sigma} \text{ (Nachfolger) definiert durch:}$$

$$S_0 = S_{\mathbb{N}}, \quad S_{\sigma \rightarrow \tau} = \lambda \alpha. \lambda \beta. (S_{\tau}(\alpha(\beta))).$$

$$3) 0_{\sigma} \in M_{\sigma} \text{ (Null) definiert durch:}$$

$$0_0 = 0_{\mathbb{N}}, \quad 0_{\sigma \rightarrow \sigma} = \lambda \alpha. \alpha, \quad 0_{(\sigma \rightarrow \tau) \rightarrow \sigma} = \lambda \alpha. (0_{\tau \rightarrow \sigma}(\alpha(0_{\tau}))),$$

$$0_{\sigma \rightarrow (\tau \rightarrow \mu)} = \lambda \alpha. \lambda \beta. (0_{\tau \rightarrow \mu}(\alpha) +_{\mu} 0_{\tau \rightarrow \mu}(\beta)).$$

$$4) E_{\sigma} \in M_{\sigma} \text{ (Eins) definiert durch } E_{\sigma} = S_{\sigma}(0_{\sigma}).$$

$$5) I_{\sigma} \in M_{\sigma \rightarrow \sigma} \text{ (Identität) definiert durch } I_{\sigma} = \lambda \alpha. \alpha.$$

Die Typindices dieser Funktoren werden im folgenden weggelassen, wenn sie ohne besonderes Interesse und durch den Zusammenhang eindeutig bestimmt sind.

LEMMA 0

- 1) $\forall \alpha \in M_{\mathcal{G}} : S_{\mathcal{G}}(\alpha) \geq \alpha$.
- 2) $\forall \alpha, \beta \in M_{\mathcal{G}} : \alpha \leq \beta \Rightarrow S_{\mathcal{G}}(\alpha) \leq S_{\mathcal{G}}(\beta)$.
- 3) $E_{\mathcal{G}} \rightarrow_{\tau} (E_{\mathcal{G}}) \geq E_{\tau}$.
- 4) $\forall \alpha \in M_{\mathcal{G}} \rightarrow_{\tau} \forall \beta \in M_{\tau} : \alpha(E_{\mathcal{G}}) \rightarrow_{\tau} \beta \geq \beta$.
- 5) $\forall \alpha \in M_{\mu} \forall \beta \in M_{\tau} : E_{\mu} \rightarrow_{\tau}(\alpha) \rightarrow_{\tau} \beta \geq \beta$.

Beweis: einfache Induktion.

BELEGUNG

Eine Belegung A ist eine eindeutige Zuordnung von monotonen Funktionalen aus $M_{\mathcal{G}}$ zu den Variablen vom Typ $\mathcal{G}$.

Das x unter A zugeordnete Funktional wird mit x^A bezeichnet.

$A_x[\alpha]$ bezeichnet diejenige Belegung, die auf allen Variablen außer x mit A übereinstimmt, und x das Funktional α zuordnet.

WERT VON TERMEN UNTER BELEGUNGEN

Sei t ein Term vom Typ $\mathcal{G}$, A eine Belegung, dann wird dem Term t das Funktional $t^A \in M_{\mathcal{G}}$ zugeordnet durch:

- 1) $x^{\mathcal{G}}$ (Variable) wird zugeordnet $x^A \in M_{\mathcal{G}}$.
- 2) $(\lambda x^{\mathcal{G}} . t^{\tau})^A = \lambda \alpha . (t^A_x[\alpha] \rightarrow_{\tau} E_{\mathcal{G} \rightarrow \tau}(\alpha))$
- 3) $(\langle u_i \rangle t)^A = \sum_{k=0}^{t^A} \left(\sum_{i=0}^k (u_i^A \rightarrow_{\mathcal{G}} E_{\mathcal{G}}) \rightarrow_{\mathcal{G}} E_{\mathcal{G}} \right)$
- 4) $(k)^A = k + E$.
- 5) $(st)^A = s^A(t^A)$.

LEMMA 1

Die Zuordnung ist wohldefiniert.

Beweis:

Triviale Induktion. Man mache sich jedoch klar, daß an dieser Stelle, in einem konstruktiven Aufbau, ein Normalformsatz eingeht: Jedem Term $\lambda x. (...)$ entspricht eindeutig ein Funktional aus M .

LEMMA 2

1) Für alle Terme t vom Typ $\mathcal{O}$ und Belegungen A gilt $t^A \geq E_{\mathcal{O}}$.

2) Der Wert t^A hängt nur von den Werten von A auf den Variablen ab, die in t frei vorkommen.

Beweis: Induktion über den Aufbau von t .

LEMMA 3

Für jede Belegung A gilt:

- 1) $((\lambda x. t)_s)^A > (t_x[s])^A$
- 2) $(\langle u_i \rangle_k)^A > u_k^A$
- 3) $((\langle u_i \rangle t)_s)^A > (\langle u_i s \rangle t)^A$
- 4) $(\langle u_i \rangle (\langle t_j \rangle s))^A > (\langle \langle u_i \rangle t_j \rangle s)^A$

Beweis:

$$1) ((\lambda x. t)_s)^A = (t_x^A[s^A] + E(s^A)) > t_x^A[s^A] = (t_x[s])^A.$$

$$2) (\langle u_i \rangle_k)^A = \sum_{i=0}^{k+1} \left(\sum_{l=0}^i (u_l^A + E) + E \right) > u_k^A.$$

$$\begin{aligned} 3) ((\langle u_i \rangle t)_s)^A &= \left(\sum_{i=0}^{t^A} \left(\sum_{l=0}^i (u_l^A + E) + E \right) s^A \right) \\ &= \sum_{i=0}^{t^A} \left(\sum_{l=0}^i (u_l^A(s^A) + E(s^A)) + E(s^A) \right) \end{aligned}$$

$$\begin{aligned}
 &\geq \sum_{i=0}^{t^A} \left(\sum_{l=0}^i (u_1^A(s^A) + E(E)) + E(E) \right) \\
 &> \sum_{i=0}^{t^A} \left(\sum_{l=0}^i (u_1^A(s^A) + E) + E \right) = (\langle u_1 s \rangle t)^A .
 \end{aligned}$$

3) Im folgenden steht $\overline{t_i}$ als Abkürzung für $\sum_{j=0}^i (t_j^A + E)$

$$\begin{aligned}
 (\langle u_1 \rangle (\langle t_j \rangle s)^A) &= \sum_{j=0}^{s^A} (\overline{t_j} + E) \sum_{k=0}^{\overline{t_j}} (\overline{u_k} + E) \\
 &= \sum_{k=0}^{\overline{t_0}} (\overline{u_k} + E) + \dots + \sum_{k=\overline{t_0}+1+\dots+\overline{t_{s-1}^A}+1}^{\overline{t_0}+\overline{t_1}+\dots+\overline{t_{s-1}^A}} (\overline{u_k} + E) + \overline{u_{\sum_{j=0}^{s-1} (\overline{t_j} + E)}} \\
 &> \sum_{j=0}^{s^A} \left(\sum_{k=0}^{\overline{t_j}} (\overline{u_k} + E) + E \right) \\
 &= \sum_{j=0}^{s^A} \left(\left(\sum_{k=0}^{t_0^A} (\overline{u_k} + E) + \dots + \sum_{k=\overline{t_0}+1+\dots+\overline{t_{j-1}^A}+1}^{t_0^A+\dots+t_{j-1}^A} (\overline{u_k} + E) + \overline{u_{\sum_{j=0}^{j-1} (\overline{t_j} + E)}} + E \right) \right) \\
 &> \sum_{j=0}^{s^A} \left(\left(\sum_{l=0}^j \left(\sum_{k=0}^{t_l^A} (\overline{u_k} + E) + E \right) + E \right) \right) \\
 &= \sum_{j=0}^{s^A} \left(\sum_{l=0}^j ((\langle u_k \rangle t_l)^A + E) + E \right) \\
 &= \sum_{j=0}^{s^A} \left((\langle u_k \rangle t_j)^A + E \right) = (\langle \langle u_k \rangle t_j \rangle s)^A
 \end{aligned}$$

Das Ziel - starke Normalisierung für geschlossene Terme vom Typ o - kann nun nicht direkt angegangen werden.

Ähnlich wie man bei Normalformsätzen mit Berechenbarkeitsprädikaten den Begriff "Berechenbar unter Substitution" braucht, muß auch hier eine allgemeinere Behauptung bewiesen werden, um die Substitution behandeln zu können.

Es werden nämlich alle Terme als Teilterme von geschlossenen Termen vom Typ o betrachtet. Dadurch sind letztlich alle im Laufe einer Reduktion möglichen Substitutionen für Variable bestimmt.

Der Teil, der den gegebenen Term vom Typ $\mathcal{G}$ zu einem geschlossenen Term vom Typ o ergänzt, heißt ein $\mathcal{G}$ -Rahmen.

RAHMEN

Ein $\mathcal{G}$ -Rahmen ist anschaulich gesagt ein geschlossener Term vom Typ o , der genau einmal das Zeichen $\mathcal{G}(x_1, \dots, x_n)$ als Stellvertreter eines Termes vom Typ $\mathcal{G}$ mit den freien Variablen $x_1, \dots, x_n$ enthält.

Als Mitteilungszeichen werden $f, g, h, \dots$ verwendet.

Im einzelnen definiert man:

$\mathcal{G}$ -Rahmen, für jeden Typ $\mathcal{G}$:

- 1) $\mathcal{G}(x_1, \dots, x_n)$ ist ein $\mathcal{G}$ -Rahmen vom Typ $\mathcal{G}$ mit freien Variablen $x_1, \dots, x_n$.
- 2) Ist f ein $\mathcal{G}$ -Rahmen vom Typ τ , x eine Variable vom Typ μ , so ist $\lambda x.f$ ein $\mathcal{G}$ -Rahmen vom Typ $\mu \rightarrow \tau$.
- 3) Ist f ein $\mathcal{G}$ -Rahmen vom Typ $\tau \rightarrow \mu$, t ein Term vom Typ τ , so ist ft ein $\mathcal{G}$ -Rahmen vom Typ μ .
- 4) Ist f ein $\mathcal{G}$ -Rahmen vom Typ τ und t ein Term vom Typ $\tau \rightarrow \mu$, so ist tf ein $\mathcal{G}$ -Rahmen vom Typ μ .
- 5) Ist f ein $\mathcal{G}$ -Rahmen vom Typ o und für alle $i \in \mathbb{N}$, u_i ein Term vom Typ τ , so ist $\langle u_i \rangle f$ ein $\mathcal{G}$ -Rahmen vom Typ τ .
- 6) Ist f ein $\mathcal{G}$ -Rahmen vom Typ τ , t ein Term vom Typ o und für alle $i \neq k$ u_i ein Term vom Typ τ , so ist $\langle u_0, \dots, u_{k-1}, f, u_{k+1}, \dots \rangle t$ ein $\mathcal{G}$ -Rahmen vom Typ τ .

Freies und gebundenes Vorkommen von Variablen in $\mathcal{G}$ -Rahmen ist wie üblich definiert.

Im folgenden verstehe ich unter einem $\mathcal{G}$ -Rahmen einen geschlossenen $\mathcal{G}$ -Rahmen vom Typ 0.

Ein $\mathcal{G}$ -Rahmen enthält genau einmal den Term $*_{\mathcal{G}}(x_1, \dots, x_n)$. Man spricht auch von einem $\mathcal{G}$ -Rahmen für die Variablen $x_1, \dots, x_n$.

Ein $\mathcal{G}$ -Rahmen f stellt kanonisch eine Funktion von Termen vom Typ $\mathcal{G}$ auf Terme vom Typ 0 dar, nämlich jedem Term t wird der Term zugeordnet, den man erhält, wenn man in f den Term $*_{\mathcal{G}}(\dots)$ durch t ersetzt. Dieser Term wird mit $f(t)$ bezeichnet.

Interessant sind hier besonders die Fälle, in denen $f(t)$ geschlossen ist; also dann, wenn f ein Rahmen für die freien Variablen von t ist. f heißt dann ein Rahmen für t .

Ganz analog stellt ein $\mathcal{G}$ -Rahmen eine Funktion von $\mathcal{T}$ -Rahmen vom Typ $\mathcal{G}$ auf $\mathcal{T}$ -Rahmen vom Typ 0 dar.

In diesem Sinne sind Gleichungen der Gestalt $f(t) = g(s)$ oder auch $f(h) = g$ (f, g, h Rahmen; t, s Terme) zu verstehen.

Ist t ein Term und f ein Rahmen für t , so ist also $f(t)$ ein geschlossener Term vom Typ 0.

$f(t)^A$ ist somit eine natürliche Zahl, die, da $f(t)$ geschlossen ist, eigentlich gar nicht von der Belegung A abhängt. Diese Zahl wird mit t^f bezeichnet.

REDUKTION

Für Terme t vom Typ $\mathcal{G}$ wird relativ zu einem $\mathcal{G}$ -Rahmen f für t die Reduktion $t \stackrel{f}{\mapsto} t'$ definiert durch:

$$2) (\lambda x.t)s \stackrel{f}{\mapsto} t_x[s] \text{ falls } ((\lambda x.t)s)^f > (t_x[s])^f$$

$$2') \langle u_i \rangle_k \stackrel{f}{\mapsto} u_k \text{ falls } (\langle u_i \rangle_k)^f > u_k^f$$

$$3) (\langle u_i \rangle t)s \stackrel{f}{\mapsto} \langle u_i s \rangle t \text{ falls } ((\langle u_i \rangle t)s)^f > (\langle u_i s \rangle t)^f$$

$$3') \langle u_i \rangle (\langle t_j \rangle s) \stackrel{f}{\mapsto} \langle \langle u_i \rangle t_j \rangle s \text{ falls } (\langle u_i \rangle (\langle t_j \rangle s))^f > (\langle \langle u_i \rangle t_j \rangle s)^f$$

$$4) t \stackrel{f}{\mapsto} t' \Rightarrow ts \stackrel{f}{\mapsto} t's \text{ falls } f(*_{\mathcal{G}}(\dots)) = g(*_{\mathcal{G}}(\dots)s).$$

$$4') s \stackrel{f}{\mapsto} s' \Rightarrow ts \stackrel{f}{\mapsto} ts' \text{ falls } f(*_{\mathcal{G}}(\dots)) = g(t*_{\mathcal{G}}(\dots)).$$

$$4'') t \stackrel{f}{\mapsto} t' \Rightarrow \lambda x.t \stackrel{f}{\mapsto} \lambda x.t' \\ \text{falls } f(*_{\mathcal{G}}(\dots)) = g(\lambda x.*_{\mathcal{G}}(\dots)).$$

$$\begin{aligned}
 4''') \quad t \stackrel{f}{\rightarrow} t' &\Rightarrow \langle u_i \rangle t \stackrel{g}{\rightarrow} \langle u_i \rangle t' \\
 &\quad \text{falls } f(*g(\dots)) = g(\langle u_i \rangle *g(\dots)). \\
 4''') \quad u_k \stackrel{f}{\rightarrow} u'_k &\Rightarrow \langle u_0, \dots, u_k, \dots \rangle t \stackrel{g}{\rightarrow} \langle u_0, \dots, u'_k, \dots \rangle t \\
 &\quad \text{falls } f(*g(\dots)) = g(\langle \dots, u_{k-1}, *g(\dots), \dots \rangle t) \\
 5) \quad t \stackrel{f}{\rightarrow} t' \text{ und } t' \stackrel{f}{\rightarrow} t'' &\Rightarrow t \stackrel{f}{\rightarrow} t'' .
 \end{aligned}$$

Vergleicht man Lemma 3 mit den Regeln 2, 2', 3, 3' sieht man den Schlüssel für den starken Normalisierungssatz: Reduktionen werden nur noch dort erlaubt, wo sie auch einen Einfluß auf die Reduktion des Termes auf seine Normalform haben.

SATZ starke Normalisierung

- 1) Ist $t \stackrel{f}{\rightarrow} t'$, so gilt $t^f > t'^f$.
- 2) Insbesondere bricht jede Kette $t_0 \stackrel{f}{\rightarrow} t_1 \stackrel{f}{\rightarrow} t_2 \stackrel{f}{\rightarrow} \dots$

nach endlich vielen Schritten ab.

Beweis:

2) folgt direkt aus 1), da $t_0^f > t_1^f > t_2^f > \dots$ eine absteigende Folge natürlicher Zahlen ist.

1) sieht man durch Induktion über die Herleitung von $t \stackrel{f}{\rightarrow} t'$ und Fallunterscheidung nach der zuletzt benutzten Regel:

Für die Regeln 2, 2', 3, 3' ist die Behauptung Bestandteil der Regel.

Für die Regeln 4, 4', 4'', 4''', 4''', 5 folgt die Behauptung induktiv:

Zum Beispiel bei 4) :

$$g(*g(\dots)s) = f(*g(\dots)) \text{ und } t^f > t'^f \Rightarrow (ts)^g = t^f > t'^f = (t's)^g.$$

oder für 5):

$$t^f > t'^f \text{ und } t'^f > t''^f \Rightarrow t^f > t''^f.$$

PAARBILDUNG

Der Satz ist noch auf einen Termkalkül mit Paarbildung erweiterbar.

Die Definition von monotonen Funktionalen wird ergänzt durch den

Fall $M_{\sigma, \tau} = \{ [\alpha, \beta] \mid \alpha \in M_{\sigma} \text{ und } \beta \in M_{\tau} \}$,

$[\alpha, \beta] < [\alpha', \beta']$ gilt genau wenn $\alpha < \alpha'$ und $\beta < \beta'$.

Die Projektionen p und q sind natürlich selbst monotone Funktionale

Die Definition von Belegungen und Rahmen übertragen sich problemlos.

Bei der Zuordnung von Funktionalen zu Termen wähle man:

$$(p(t))^A = p(t^A) + E; \quad (q(t))^A = q(t^A) + E;$$

$$([\![t, s]\!])^A = [\![t^A + E(s^A), s^A + E(t^A)]\!].$$

VERHÄLTNISS VON $f \vdash$ ZU $\vdash$

Unter $\vdash$, der Standardreduktion verstehe ich hier die in Kap I.2. definierte Reduktion $\vdash$ für den obigen Kalkül $T_{\sigma\sigma}$.

Das Verhältniss von $f \vdash$ zu $\vdash$ ist entscheidend für den Aussage-wert des obigen starken Normalisierungssatzes, da doch gezeigt werden muß, daß die Einschränkungen bei den Regeln 2, 2', 3 und 3' nicht zu streng sind, sondern gerade nötig um eine starke Normalisierung zu erreichen.

Zunächst ist klar, daß $t \stackrel{f}{\vdash} t' \Rightarrow t \vdash t'$. Dies läßt sich induktiv über die Herleitung von $t \stackrel{f}{\vdash} t'$ zeigen.

Die Umkehrung dieser Aussage lautet: Gilt $t \vdash t'$ und ist t' normal vom Typ σ und geschlossen, so gilt $t \stackrel{f}{\vdash} t'$ wobei $f = *_0()$ der leere σ -Rahmen ist.

Der Beweis dieser Behauptung sei hier in einer etwas informelleren Form geführt um allzuviel technischen Formalismus zu vermeiden.

Sei nun eine Herleitung von $t \vdash t'$ gegeben. Diese Herleitung kann folgendermaßen schrittweise in eine Herleitung $t \stackrel{f}{\vdash} t'$ umgeformt werden. (Dabei ist $f = *_0()$ der leere σ -Rahmen):

Zuerst behandelt man die Unterschiede bei den Verträglichkeits-Regeln. Die allgemeine Regel 4) des Kalküls $\vdash$ kann unter Zuhilfe-

nahme der Transitivität problemlos durch die Regeln 4,4',4'',4''' und anstatt 4''' einer Regel $\hat{4}$ ersetzt werden. (Wobei die Rahmen noch unberücksichtigt bleiben.)

$\hat{4}$ ist dabei:

$$\hat{4) \quad u_i \stackrel{f}{\vdash} u'_i \quad \text{für alle } i \in \mathbb{N} \quad \Rightarrow \quad \langle u_i \rangle_t \stackrel{f}{\vdash} \langle u'_i \rangle_t$$

$$\text{falls } f(\langle u'_0, u'_1, \dots, u'_{i-1}, *(\dots), u_{i+1}, u_{i+2}, \dots \rangle_t) = f_i(*(\dots))$$

$$\text{für alle } i \in \mathbb{N}.$$

Als nächstes muß man die Herleitungsregeln mit geeigneten Rahmen versehen.

Der Rahmen an der Wurzel der Herleitung liegt fest: $*_0()$.

Nun ergibt sich bei jeder Regel des Kalküls $\mathcal{E}$ aus dem Rahmen für die Konklusion in eindeutiger Weise ein Rahmen für die Prämissen, wegen der Nebenbedingungen an die Rahmen.

Die Ausstattung mit Rahmen kann also induktiv von der Wurzel aus geschehen.

Die Grundidee der folgenden Ausführungen ist es, daß echte Reduktionen

$u \stackrel{f}{\vdash} u'$ mit $u \stackrel{f}{\leq} u'$ in unwichtigen Teiltermen von $f(u)$ stattfinden und problemlos gestrichen werden können. Danach bleiben für die Regel $\hat{4}$ nur noch endlich viele Prämissen, sodaß sie durch die Regel 4''' ersetzt werden kann.

RAHMEN ALS FUNKTIONALE

Sei f ein $\mathcal{G}$ -Rahmen vom Typ τ , für die Variablen $x_1, \dots, x_n$ vom Typ $\mathcal{G}_1, \dots, \mathcal{G}_n$ und A eine Belegung, dann ist das Funktional

$f^A \in F(\mathcal{G}_1 \rightarrow \dots \rightarrow \mathcal{G}_n \rightarrow \mathcal{G}) \rightarrow \tau$ definiert durch:

- a) $f = *_0(x_1, \dots, x_n) \quad \Rightarrow \quad f^A = \lambda \alpha. \alpha(x_1^A, \dots, x_n^A)$
- b) $f = tg \quad \Rightarrow \quad f^A = \lambda \alpha. t^A(g^A \alpha)$
- c) $f = gt \quad \Rightarrow \quad f^A = \lambda \alpha. ((g^A \alpha) t^A)$
- d) $f = \lambda x. g \quad \Rightarrow \quad f^A = \lambda \alpha. \lambda \beta. (g^A x [\beta] \alpha + E(\beta))$

$$e) \quad f = \langle t_i \rangle g \quad \Rightarrow \quad f^A = \lambda \alpha \cdot \sum_{k=0}^{g^A \alpha} \left(\sum_{i=0}^k (t_i^A + E) + E \right)$$

$$f) \quad f = \langle \dots, t_{j-1}, g, t_{j+1}, \dots \rangle t$$

$$\Rightarrow \quad f^A = \lambda \alpha \cdot \sum_{k=0}^{t^A} \left(\sum_{i=0}^k (\beta_i + E) + E \right)$$

$$\text{mit } \beta_i = t_i^A \quad \text{für } i \neq j$$

$$\beta_i = g^A \alpha \quad \text{für } i = j$$

dabei sind f, g Rahmen und t, t_i Terme.

LEMMA 4

1) f^A hängt nur von den Werten von A auf den Variablen ab, die frei sind in f .

2) f^A ist entweder konstant $f^A = \lambda \alpha \cdot \beta \quad \beta \in M_\tau$
oder monoton $f^A \in M(\sigma_1 \rightarrow \dots \rightarrow \sigma_n \rightarrow \sigma) \rightarrow \tau$.

(Der Fall einer konstanten Funktion tritt durch die Regel f) hinzu, für $t^A < j$.)

Beweis: Induktion nach dem Aufbau von f .

Bemerkung;

Inhaltlich ist f^A gerade das Funktional $t^A \mapsto f(t)^A$, jedoch für allgemeine $\alpha \in M_\sigma$ definiert, statt für spezielle Werte von der Form $t^A \in M_\sigma$.

Der Zweck ist die folgende Definition von wichtigen und unwichtigen Teiltermen in dem Sinne, daß Reduktionen nur in wichtigen Teiltermen eine Rolle spielen.

WICHTIGE TEILTERME

Sei t ein geschlossener Term und s ein Teilterm von t , weiter seien $x_1, \dots, x_n$ die freien Variablen von s (in t gebunden).

Sei dann f der Rahmen, der entsteht, wenn man in t den Teilterm s durch $f(x_1, \dots, x_n)$ ersetzt. f ist dann ein geschlossener Rahmen, so daß f^A nicht von der gewählten Belegung A abhängt.

Ist dann f^A konstant so ist s ein unwichtiger andernfalls ein wichtiger Teilterm von t .

Der entscheidende Schritt ist nun die Umformung der Regel $\hat{4}$ in die endliche Regel $4'''$. Dies muß so geschehen, daß die Nebenbedingungen der Regeln $2, 2', 3, 3'$ erfüllt sind.

man streicht dazu zuerst alle Anwendungen der Regeln

$(2, 2', 3, 3')$ $u \xrightarrow{f} u'$ falls $u^f \leq u'^f$ und ersetzt sie durch die neue Regel 1 :

1) $u \xrightarrow{f} u$.

Die Substitution von u' durch u hat natürlich weitere Konsequenzen. für die Herleitung. Um die Herleitung wieder korrekt zu machen, sind weitere Substitutionen nötig, die sukzessive vorgenommen werden können. Anschaulich geschieht dies von links unten nach oben und von rechts oben nach unten in den Herleitungen.

Die exakte Beschreibung sei nun skizziert:

Wird eine Regel $4, 4', 4'', 4'''$ $\frac{u \xrightarrow{f} u'}{v \xrightarrow{g} v'}$ durch Substitution $u' \rightarrow u''$

geändert, so gibt es eindeutig ein v'' , sodaß $\frac{u \xrightarrow{f} u''}{v \xrightarrow{g} v''}$ eine

korrekte Anwendung derselben Regel ist.

Wird sie durch Substitution in v und g geändert $v \rightarrow v'', g \rightarrow g''$,

existiert ebenso ein u'' und f'' daß $\frac{u'' \xrightarrow{f''} u'}{v'' \xrightarrow{g''} v'}$ korrekt ist.

Die Regel 4 wird nur insofern besonders behandelt, als sie mehrere Prämissen haben kann.

Wird eine Regel 5) $\frac{u \xrightarrow{f} u' \quad u' \xrightarrow{f} v}{u \xrightarrow{f} v}$ durch eine

Substitution $u' \rightarrow u''$ in der linken Prämisse geändert entsteht

durch dieselbe Substitution in der rechten Prämisse wieder eine korrekte Regel. Ebenso wird eine Substitution $v \rightarrow v''$ in der rechten Prämisse auf die Konklusion übertragen.

Eine Substitution $u \rightarrow u''$ und $f \rightarrow f''$ in der Konklusion wird analog wie bei den Verträglichkeitsregeln auf die Prämissen übertragen:

$$5) \quad \frac{u \stackrel{f''}{\vdash} u' \quad u' \stackrel{f''}{\vdash} v''}{u \stackrel{f''}{\vdash} v''}$$

Wird eine Regel 2, 2', 3, 3' $u \stackrel{f}{\vdash} v$ durch Substitution $u \rightarrow u'$ und $f \rightarrow f'$ geändert, so gibt es eindeutig ein v' so, daß $u' \stackrel{f'}{\vdash} v'$ eine korrekte Regelanwendung derselben Regel ist, falls die Substitution nicht die äußere Form $B(K(\dots)\dots)$ von u Bbetrifft (B Beseitigung, K kritische Beseitigung oder Einführung). Andernfalls wählt man $v' = u'$ (falls ganz u substituiert wird) und erhält eine Anwendung der Regel 1).

Als Ergebnisse erhält man eine Herleitung, bei der für alle Regeln 2, 2', 3, 3' die Nebenbedingungen erfüllt sind.

Nun streicht man alle Regeln 1) zusammen mit den folgenden Verträglichkeitsregeln bis man auf eine Prämisse der Regel $\hat{4}$ oder 5 kommt.

Da nur Herleitungen von Identitäten gestrichen werden wird im Falle der Regel 5 die Regelanwendung selbst überflüssig (Konklusion = zweite Prämisse) und im Falle der Regel $\hat{4}$ kann die Prämisse einfach entfallen. Ist es aber die letzte Prämisse der Regel $\hat{4}$ gewesen so wird das Streichen bei der Konklusion fortgesetzt.

Es ist nun zu bemerken, daß durch oben beschriebene Substitutionen $u \stackrel{f}{\vdash} v \rightarrow u' \stackrel{f'}{\vdash} v'$ nur unwichtige Teilterme von $f(u)$, bzw. $f(v)$ geändert werden (Dies folgt sofort induktiv aus der Definition der Substitutionen. Lemma 3 ist Induktionsanfang.) und außerdem nur nach "vorne" in der Herleitung.

Damit folgt sofort, daß die Endformel $t \stackrel{f_i}{\vdash} k$ unverändert bleibt.

Für die verbliebenen Prämissen $u_i \stackrel{f_i}{\vdash} u_i$ der Regel 4 folgt durch Induktion über die Herleitung direkt, daß $u_i \stackrel{f_i}{\vdash} u_i^{f_i}$

Damit kann die Regel $\hat{4}$ aber nur noch endlich viele Prämissen besitzen, denn die $u_i \stackrel{f_i}{\vdash}$ bilden eine echt absteigende Folge natürlicher Zahlen. Eine Regel $\hat{4}$, die nur endlich viele Prämissen besitzt, kann aber mit Hilfe der Transitivität problemlos in eine Anzahl von Regeln 4"" übersetzt werden.

3. Stetige Normalisierung

a) Vorbemerkungen und Definition

Der Begriff der stetigen Normalisierung läßt sich am leichtesten erklären, wenn man Terme als Funktionen auf Bäumen betrachtet. Ein Funktional F , das einen Term t auf einen Term $F(t)$ abbildet, heißt dann stetig, wenn der Wert von $F(t)$ an einer bestimmten Stelle des Baumes nur von endlich vielen Werten von t abhängt. (Eine Umgebung eines Termes wird bestimmt durch alle Terme, die auf endlich vielen endlichen Anfangsstücken (Wege zur Wurzel) des Baumes mit ihm übereinstimmen.)

Auf stetige Normalisierungen kommt man zwangsweise, wenn man etwa rekursive Normalisierungen betrachtet, die natürlich stetig sind. Schließlich haben stetige Normalisierungen den Vorteil, daß sie auf beliebigen Termen definiert sind, also totale Funktionale sind, während üblicherweise Normalisierungen nur partielle Funktionale ergeben.

So ist der hier definierte Operator N auch auf Terme anwendbar, die nach den Termbildungsregeln lokal korrekt aufgebaut sind, aber nicht fundiert sind. Beispiele für das Auftreten solcher Fälle sind in [4] zu finden.

Nun zunächst einige Definitionen zu Bäumen:

ENDLICHE FOLGEN

Mit den Buchstaben $a, b, c, \dots$ werden im folgenden endliche Folgen von natürlichen Zahlen bezeichnet. Es wird die Schreibweise

$\langle a_1, \dots, a_n \rangle$ verwendet.

Die leere Folge wird als Δ notiert.

Es werden die üblichen Operationen und Funktionen verwendet:

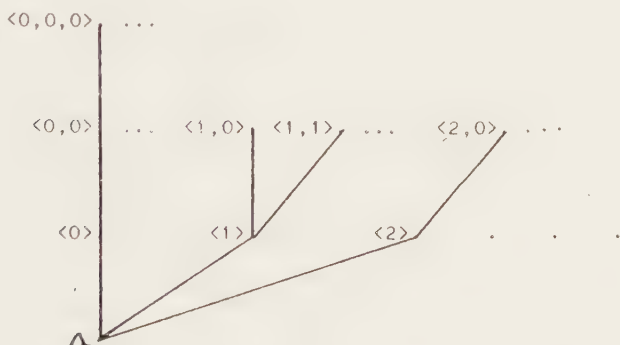
$(a)_i$ der i -te Term der Folge

$a * i$ Fortsetzung von a durch die Zahl i ,

- $a * b$ Fortsetzung von a durch die Folge b ,
 $a \subset b$ a ist Anfangsstück von b ,
 $l(a)$ Länge von a ($l(\Lambda) = 0$, $l(\langle a_1, \dots, a_n \rangle) = n$).

BÄUME und KNOTEN

Die Menge aller endlichen Folgen natürlicher Zahlen geordnet durch die Relation $\subset$ bildet den universalen Baum:



Eine endliche Folge wird deshalb auch als Knoten bezeichnet. Im allgemeinen ist jede Teilmenge der endlichen Folgen, die mit jeder Folge auch alle ihre Anfangsstücke enthält, ein Baum. Statt mit solchen Mengen direkt zu operieren werden im folgenden ihre charakteristischen Funktionen benutzt. D.h., es werden Funktionen auf dem universalen Baum betrachtet, deren Wertebereich das Zeichen $\emptyset$ enthält. Der Funktionswert $\emptyset$ wird auf allen Knoten angenommen die nicht zum betrachteten Baum gehören. Ist t ein Baum (also eine charakteristische Funktion) so wird der Wert von t am Knoten a , der Inhalt des Knotens a , mit t_a bezeichnet.

Der Teilbaum von t , der sich über dem Knoten a erstreckt, also die Menge aller c mit $a \subset c$ wird mit t_{a*} bezeichnet, sodaß $(t_{a*})_d = t_a * d$.

NOTATIONSVEREINFACHUNG

Da im folgenden sehr häufig Ausdrücke wie $t_{\langle 0 \rangle}$, $t_{\langle 0,0 \rangle}$, $N_{\langle i \rangle}$, $t_{\langle 0 \rangle *}$ etc. auftreten, wird eine Notationsvereinfachung benutzt:

Wenn endliche Folgen als untere Indices auftreten, werden einfach die Klammern "<" und ">" sowie das Komma "," weggelassen.

Obige Ausdrücke schreiben sich dann t_0 , t_{00} , N_i , t_{0*} . Aus dem Zusammenhang sollte dabei stets klar sein um was es sich handelt.

TERME

Dem folgenden liege wieder ein fester Termkalkül zugrunde, wie er mit den Methoden von Kap.I definierbar ist.

Terme können nun in natürlicher Weise als Bäume aufgefasst werden. Nämlich als Abbildung der Menge aller Folgen in die Menge der Bildungsregeln (Formen) plus das Zeichen $\emptyset$, plus Variable.

Neben diesen Werten wird nur noch ein weiterer Wert hinzugenommen, der für die Beschreibung von Termen eigentlich absolut entbehrlich ist: der Wert "Rep" (von Repetition) .

Die Einführung dieses Wertes ermöglicht einen Verzögerungsmechanismus durch Wiederholung von Werten, wie er aus der Rekursionstheorie bekannt ist. Die Anwendung des Wertes Rep wird unten noch ausführlich beschrieben.

Das Zeichen $\emptyset$ wird im folgenden auch als Bezeichnung für den leeren Term verwendet, d.h. die konstante Abbildung, die nur den Wert $\emptyset$ annimmt.

Zusätzlich zu den Formen kann (und muß) der Funktionswert natürlich noch weitere Informationen enthalten, wie etwa Angaben darüber, welche Variablen wo gebunden werden.

Die genaue Definition von Term als Funktion auf Bäumen kann leicht induktiv gegeben werden. Da die Details einer solchen Definition hier aber nicht benötigt werden, genüge statt dessen ein Beispiel:



Es ist also $t_{\lambda} = \text{App}$ oder $t_1 = z$.

STETIGE NORMALISIERUNG

Das Problem der stetigen Normalisierung kann nun so formuliert werden:
Gegeben ein Term t gesucht ein Term $N(t)$ so, daß

- a) $t \triangleright N(t)$ normal und
- b) die Berechnung von $N(t)$ an einem bestimmten Knoten a so vor sich geht, daß dabei t nur an endlich vielen Knoten berechnet werden muß (Stetigkeit).

Das im folgenden beschriebene Funktional N leistet dies.

HEURISTIK

Zum Verständniss des im folgenden Beschriebenen Funktional N sind vielleicht einige Vorbemerkungen nützlich:

Das Funktional N entstand nach einer Analyse des Verfahrens von Mints in "Finite Investigations of Transfinite Derivations" [7] zur Normalisierung von Beweisbäumen. Dieses Verfahren funktioniert grob so: Dem zu berechnenden Knoten a entspricht ein Knoten a^* im alten Baum. Die Zuordnung zwischen a und a^* ist dabei durch eine Rekursion definiert, die durch die Länge $l(a)$ beschränkt ist. Der Inhalt des Knotens a ist nur eine Modifikation des Inhalts von Knoten a^* . Diese Modifikationen bestimmen sich wieder ausschließlich durch die Inhalte von Knoten unterhalb von a^* . Damit ergibt sich Stetigkeit. Die Rep-Regel wird von Mints benötigt, damit die erwähnte Zuordnung zwischen a und a^* Kalmar-elementar wird.

Betrachtet man z.B. Mints's Regel zur Cut-Elimination
etwa:

$$\begin{array}{c}
 \vdots \\
 \frac{\Sigma \rightarrow A[n]}{\Sigma \rightarrow A[n]} \text{ (Rep)} \\
 \vdots \\
 \frac{\Sigma \rightarrow A[n] \quad A[n], \forall x:A, \Delta \rightarrow D}{\Sigma \rightarrow \forall x:A} \quad \frac{A[n], \forall x:A, \Delta \rightarrow D}{\forall x:A, \Delta \rightarrow D} \quad \frac{\Gamma, \Pi \rightarrow A[n] \quad A[n], \Gamma, \Pi, \Delta \rightarrow D}{\Gamma, \Pi, \Delta \rightarrow D} \text{ (Cut)} \\
 \vdots \\
 \Rightarrow \\
 \frac{\Gamma, \Pi \rightarrow \forall x:A \quad \forall x:A, \Gamma, \Phi \rightarrow C}{\Gamma, \Pi, \Phi \rightarrow C} \text{ (Cut)} \quad \frac{\Gamma, \Pi, \Phi \rightarrow C}{\Gamma, \Pi, \Phi \rightarrow C} \text{ (Rep)}
 \end{array}$$

so sieht man die Funktion der Rep Regel:

Sie erhält nämlich die Zuordnung zwischen alten und neuen Knoten in der Weise, daß die alten Knoten nie länger als die neuen Knoten werden. So wäre es wohl das nächstliegende, in der rechten Herleitung an der Wurzel statt Rep die Regel zu nehmen, die in der linken Herleitung über der rechten Prämisse steht. Was aber, wenn diese Regel wieder ein Cut ist? Dies würde zu einer Wiederholung des Verfahrens führen ... ad infinitum. Die Rep Regel bietet hier einen Ausweg. Geht man nun umgekehrt davon aus, daß bei einem stetigen Verfahren die Länge der alten Knoten notwendig irgendwie durch die Länge der neuen Knoten beschränkt sein muß, so folgt eine neue Interpretation der Rep-Regel: Sie drückt ein Unwissen über den Inhalt des Knotens aus, das daher kommt, daß man den alten Baum nur zu einem Teil "sehen" kann. Dieses Phänomen ist gut bekannt, es tritt etwa auch auf wenn man zeigt, daß eine Menge entscheidbar ist falls sie und ihr Komplement aufzählbar sind.

Von dieser Funktion der Rep-Regel ausgehend, kann man nun - unter Aufgabe des Zusammenhangs zwischen alten und neuen Knoten - eine einfache Maschine konstruieren, die sich ein beschränktes Anfangsstück des alten Baumes t ansieht und daraus den Wurzelknoten (Δ -Knoten) des neuen Baumes $N(t)$ berechnet. (Maschinen sind vielleicht anschaulicher als Funktionale, weswegen ich hier von Maschinen spreche. Da sich aber Funktionale leichter hinschreiben

lassen als Maschinen, werde ich später wieder zur Beschreibung mittels Funktionalen zurückkehren.) Ist der Wurzelknoten aus dem betrachteten Stück des Baumes nicht eindeutig bestimmbar, so wird der Rep-Knoten genommen. Es zeigt sich, daß schon die Knoten Λ und $\langle 0 \rangle$ des alten Baumes für unseren Zweck genügen. Versucht man nun in der selben Weise, auch höhere Knoten (etwa $N(t)_{\langle 0 \rangle}$) zu berechnen, ergibt es sich als das Einfachste, wenn man erst einmal die unteren Knoten berechnet (rekursiv), dabei die aus der Wahl dieser tieferen Knoten sich ergebenden Umordnungen des darüber sich erstreckenden Baumes mit verfolgt, um schließlich den letzten Knoten in bekannter Weise aus einem Anfangsstück dieses umgeordneten Baumes zu berechnen. Das heißt ausgehend von t_Λ und t_0 wird nicht nur der Λ -Knoten von $N(t)$ definiert, sondern auch Bäume $N_1(t)$ so, daß der nächste Knoten $N(t)_{\langle i \rangle}$ als $N(N_1(t))_\Lambda$ definiert werden kann. Dieses Verfahren läßt sich dann fortsetzen.

So kommt man endlich zu der Vorstellung einer Maschine, die sich schrittweise, Knoten für Knoten, an der Wurzel des Baumes beginnend, nach oben arbeitet und bei jedem Schritt

- zuerst ein Stück des über ihr liegenden Baumes anschaut
- daraus die Entscheidung trifft welcher Knoten als nächster kommt
- und zuletzt noch die Äste des alten Baumes "umordnet" und so über den neuen Knoten setzt daß ein korrekter Baum entsteht, der mit dem alten "gleichbedeutend" ist.

Damit ist die Ausgangsposition für den nächsten Schritt erreicht.

Man sieht also die Maschine immer über einem Stamm von schon normalen Knoten, über dem sich ein Gestrüpp von noch nicht normalen Ästen erstreckt.

Wichtig ist vielleicht noch der Gesichtspunkt, daß die Maschine natürlich nicht die über ihr liegenden "umgeordneten" Äste alle berechnet, sondern sich nur zu merken braucht, wie im Bedarfsfalle ein Knoten aus solch einem Ast berechnet werden kann (siehe auch die Anmerkungen zu den Einsetzungen bei echten Kontraktionen weiter unten).

Ein weiterer Punkt ist der, daß beim Berechnen von $N(t)$ manchmal gewisse Teilterme von t schon durchsucht wurden und festgestellt wurde, daß sie schon normal sind. Diese Information muß für spätere Berechnungen in Form einer "Markierung" der Knoten aufgehoben werden. Dies sei im folgenden exakt beschrieben.

FUNKTIONALE N und N_a

Das Funktional N bildet einen Term t ab auf einen Term $N(t)$, den zu t gehörigen normalen Term.

Die Funktionalen N_a ($\mathbf{N}$ in Fettdruck) definieren zu jedem Term t und Knoten $a \vdash \Delta$ den Term $N_a(t)$. $N_a * i(t)$ ist anschaulich der i -te Ast, der nach der Berechnung von $N(t)_a$ über dem Knoten a steht. N und N_a werden induktiv definiert.

Für die Definition wird fallweise eine zusätzliche Information über Inhalte von Knoten gebraucht, nämlich ob der Knoten, d.h. genauer der über dem Knoten sich befindende Baum, schon normal ist. Diese Informationen fallen während den rekursiven Berechnungen bisweilen an und werden an anderen Stellen auch wieder benötigt. Die Definition von Term wird in dieser Hinsicht also erweitert. Enthält ein Knoteninhalte die Information "normal" so wird dies im folgenden durch Unterstreichen gekennzeichnet. x oder Rep etc... gehören also zum Wertebereich von Termen.

DEFINITION von $N(t)_\Delta$ und $N_i(t)$

Bemerkung:

Bei einzelnen Fällen der Definition wurde zur Veranschaulichung je ein Diagramm angefügt. Es bezieht sich beispielhaft auf den Kalkül T_∞ .

Die Diagramme sind so aufgebaut, daß links der Term t steht und rechts ein Baum mit Wurzel $N(t)$ und darüber als i -ten Ast $N_i(t)$. Man erinnere sich dabei vielleicht an die oben skizzierte Maschine.

Fallunterscheidung nach t_Δ :

t_Δ = Variable x , Einführung E , E , normale Beseitigung B , Rep, Rep oder $\emptyset$.

$\rightarrow N(t)_\Delta := t_\Delta$ und $N_i(t) := t_i *$ für alle i .

t_Δ = nicht normale Beseitigung.

Fallunterscheidung nach t_0 (Hauptteil):

t_0 === Variable-oder-nicht-kritische-normale-Beseitigung ==

$\Rightarrow N(t)_{\Delta} := \underline{t_{\Delta}}$ und $N_i(t) := t_i^*$ für alle i .



t_0 === nicht-kritische-Beseitigung ==

$\Rightarrow N(t)_{\Delta} := \underline{\text{Rep}}$ und $N_0(t)_{\Delta} := t_{\Delta}$, $N_0(t)_{0*} := N(t_{0*})$
 $N_0(t)_{i*} := t_i^*$ für alle $i > 0$.
 $N_i(t) := \emptyset$ für alle $i > 0$.



Zu bemerken ist, daß hier $N_i(t)$ unter Rückgriff auf das Funktional N definiert wird. N wird nun selbst wieder (siehe unten) unter Bezug auf $N_i(t)$ definiert. Die Wohldefiniertheit von N und N ist daher nicht selbstverständlich.

Anschaulich bedeutet der Rückgriff hier, daß der Bereich des Termes noch nicht ausreicht, um den nächsten Knoten in $N(t)$ festzulegen.

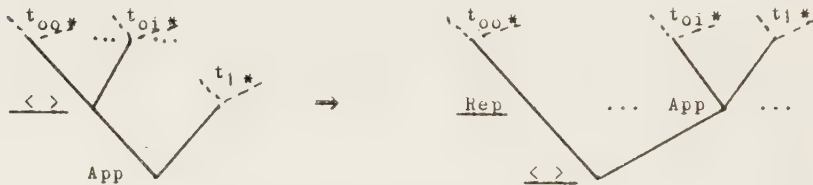
t_0 === Rep-Rep ==

$\Rightarrow N(t)_{\Delta} := \underline{\text{Rep}}$ und $N_0(t)_{\Delta} := t_{\Delta}$, $N_0(t)_{0*} := t_{00*}$
 $N_0(t)_{i*} := t_i^*$ für alle $i > 0$.
 $N_i(t) := \emptyset$ für alle $i > 0$.



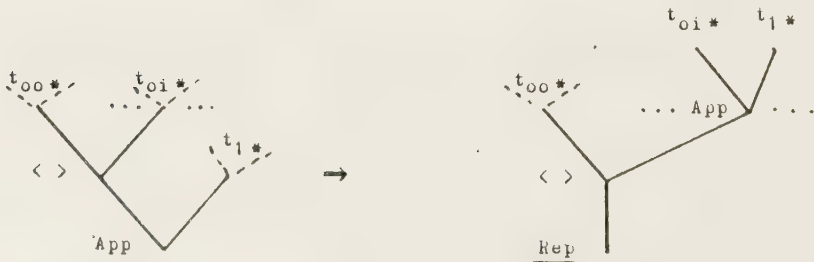
t_0 --- normale kritische Beseitigung ---

$\Rightarrow N(t)_{\Delta} := \underline{t_0}$ und $N_0(t)_{\Delta} := \underline{Rep}$, $N_0(t)_{0*} := t_{00*}$
 $M_i(t)_{\Delta} := t_{\Delta}$ für alle $i > 0$.
 $M_i(t)_{0*} := t_{0i*}$ und
 $M_i(t)_{j*} := t_{j*}$ für alle $i, j > 0$.



t_0 --- nicht normale kritische Beseitigung ---

$\Rightarrow N(t)_{\Delta} := \underline{Rep}$ und $N_0(t)_{\Delta} := t_0$, $M_0(t)_{0*} := t_{00*}$
 $M_0(t)_i := t_{\Delta}$ für alle $i > 0$.
 $M_0(t)_{i0*} := t_{0i*}$ und
 $M_0(t)_{ij*} := t_{j*}$ für alle $i, j > 0$.
 $M_i(t) := \emptyset$ für alle $i > 0$.



t_0 == Einführung ==

Sei $H = B(E(\dots, (\vec{x}_1)T_1(\vec{x}_1), \dots), \dots, (\vec{y}_j)S_j(\vec{y}_j), \dots)$, H' die zur Beseitigung $B = t_{\Delta}$ und Einführung $E = t_0$ gehörige Kontraktionsvorschrift.

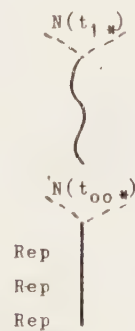
$\Rightarrow N(t)_{\Delta} := \text{Rep}$ und

$N_0(t)_{\Delta} := \text{Rep}$

$N_0(t)_{0*} := H' \dots T_1[(\vec{x}_1)\text{Rep}(N(t_{0i*}))] \dots S_j[(\vec{y}_j)N(t_{j*})]$

$N_i(t) := \emptyset$ für alle $i > 0$.

Die in einem Knoten von $N_0(t)$ eventuell vorhandene Information "normal" (Unterstreichnung) wird gelöscht.



Bemerkung:

1) Die in der Kontraktionsvorschrift verlangten Einsetzungen werden natürlich ebenso wie die Normalisierungen nicht direkt ausgeführt, sondern nur als Berechnungsvorschrift festgehalten.

Ein Term $t_x[s]$ ist als Berechnungsvorschrift auf den Knoten folgendermaßen zu lesen: Man berechnet $t_x[s]$ an der Stelle a , indem man t an allen Anfangsstücken a' von a berechnet. Gilt dann für ein solches a' mit $a = a' * c$, daß $t_{a'} = x$, frei in t , so ist $t_x[s]_a = s_c$ andernfalls ist $t_x[s]_a = t_a$.

2) $\text{Rep}(s)$ ist definiert durch

$\text{Rep}(s)_{\Delta} = \text{Rep}$, $\text{Rep}(s)_{i*} = \emptyset$ $i > 0$, $\text{Rep}(s)_{0*} = s_b$.

3) Für einen reinen Normalisierungssatz kann darauf verzichtet werden, auf alle Teilterme das Funktional N anzuwenden, ebenso kann der Einschub des Rep 's bei der Substitution unterbleiben.

Gebraucht wird diese Definition erst beim Beweis der Behauptung, daß getypte und fundierte Terme beim Normalisieren wieder in fundierte Terme übergehen (SATZ 3). Prinzipiell ist es sogar denkbar, beim Beweis von Satz 3 ebenfalls ohne diese speziellen Forderungen auszukommen.

Die angegebene Formulierung wurde gewählt, da sie die Reduktionsstrategie "innermost"-Reduktion erzwingt. Für diesen Fall läßt sich der Beweis aus Kap III.1 (maximal innermost Reduktion) für die Fundiertheit von $N(t)$ übertragen.

Für das folgende ist es nützlich $N_{\Delta}(t) := t$ zu setzen.

REKURSIONSGLEICHUNG

Es ist nun für alle a :

$$N(t)_a = N(\mathbb{N}_a(t))_{\Delta}$$

und

$$\mathbb{N}_{a * i}(t) = \mathbb{N}_i(\mathbb{N}_a(t)) \quad \text{für alle } i \in \mathbb{N}.$$

LEMMA I

Für alle Knoten a, b, c gilt :

$$1) \quad N(t)_{a * b} = N(\mathbb{N}_a(t))_b$$

$$2) \quad \mathbb{N}_c(\mathbb{N}_a(t)) = \mathbb{N}_{a * c}(t)$$

Beweis:

Induktion nach $l(c)$ und $l(b)$:

$l(c) = 0$ oder 1 und $l(b) = 0$:

Die Behauptung ist trivial bzw. gerade die Rekursionsgleichung.

Induktionsschritt:

Sei nun die Behauptung 1) bewiesen für alle b mit $l(b) < n$ und Behauptung 2) für alle c mit $l(c) < n+1$. Es folgt mit den Rekursionsgleichungen und der IV für b, c mit $l(c) = n+1$ und $l(b) = n$:

$$N(t)_{a * b} = N(\mathbb{N}_{a * b}(t))_{\Delta} = N(\mathbb{N}_b(\mathbb{N}_a(t)))_{\Delta} = N(\mathbb{N}_a(t))_b$$

Sei $c = c' * i$:

$$N_{c' * i}(N_a(t)) = N_i(N_{c'}(N_a(t))) = N_i(N_a * c'(t)) = N_a * c' * i(t) .$$

Nachdem nun für N und N Rekursionsgleichungen gegeben sind, sowie definierende Gleichungen für $N(t)$ und $N_i(t)$, stellt sich als erstes die Frage, ob dadurch N und N wohldefiniert ist. D.h. Ob die gegebenen Rekursionsgleichungen als Berechnungsvorschrift ausreichen. Fragen nach der Berechnungskomplexität, der Stetigkeit und der Normalität schließen sich an.

WOHLDEFINIERTHEIT

Durch Inspektion der verschiedenen Fälle bei der Definition von $N(t)_A$ und $N_i(t)$ sieht man:

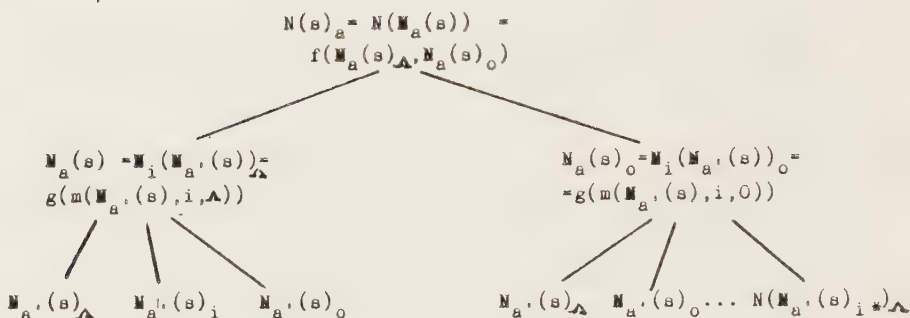
- 1) Für einen Term s wird $N(s)_A$ durch eine Fallunterscheidung nach s_A und s_0 explizit angegeben. D.h. $N(s)_A$ wird durch eine elementare Funktion f gegeben als $N(s)_A = f(s_A, s_0)$.
- 2) Für einen Term s , $i \in \mathbb{N}$ und einen Knoten b ist $N_i(s)_b$ definiert zuerst durch Fallunterscheidung nach s_A und s_0 .

Die Berechnung von $N_i(s)_b$ erfordert je nach den verschiedenen Fällen die Auswertung von Termen der Form:

$$\begin{aligned} & s_c \text{ (c definiert durch b mit } l(c) \leq l(b)+1) \\ \text{oder} \quad & N(s')_c \text{ mit } s' = s_j * , l(c)+1 \leq l(b) \\ & \text{oder } s' = s_{0j} * , l(c)+2 \leq l(b) \\ & \text{wobei j und c definiert durch i, b und s.} \end{aligned}$$

$N_i(s)_b$ wird also gegeben durch eine elementare Funktion g auf einer endlichen Menge $m(s, i, b)$ von Knotenwerten s_c und $N(s')_c$.
 $N_i(s)_b = g(m(s, i, b))$ ist also elementar.

Durch die Rekursionsgleichungen erhält man für beliebige Knoten a für $N(s)_a$ einen "Berechnungsbaum" der Form:



Jeder Term im Baum, der nicht schon die Gestalt t_c oder $N(\dots)\Delta$ oder $M_1(\dots)_b$ hat, läßt sich durch Anwendung der Rekursionsgleichungen in eine solche Form bringen. Er kann dann wieder als elementare Funktion f, g von -wie gezeigt wird- "kleineren" Ausdrücken geschrieben werden, was den Baum nach unten fortsetzt.

Ein Berechnungsbaum für $N(t)$ enthält also Berechnungsvorschriften, die aus den Symbolen t (Term), N , M , und Knoten aufgebaut sind.

Alle Endknoten im Baum haben die Form t_c .

Die Wohldefiniertheit von N und M folgt damit sofort, wenn klar ist, daß diese Berechnungsbäume nicht unbeschränkt weiter wachsen.

In der Tat wird gezeigt, daß die Höhe des Berechnungsbaumes für $N(t)_a$ höchstens $2^{l(a)+1} - 1$ beträgt.

Die Stetigkeit folgt dann, da der Berechnungsbaum nur endliche Verzweigungen besitzt, (die von den speziellen Kontraktionsschemata abhängen) und bei einer endlichen Höhe also nur endlich viele Endknoten der Form t_c besitzen kann. Der Wert von $N(t)_a$ hängt aber nur von diesen Werten t_c ab.

HÖHE VON BERECHNUNGSBÄUMEN

Um eine Abschätzung der Höhe dieser Berechnungsbäume (eine Art Rekursionstiefe) angeben zu können, ist es nötig, für beliebige Ausdrücke aus N , $\mathbb{N}$, t , etc. wie sie im Baum auftreten können, eine Abschätzung für die Höhe ihres Berechnungsbaums zu haben. Um diese Abschätzungen zu gewinnen, werden "Berechnungsterme" eingeführt. Sie reflektieren die Terme im Baum, was den Aufbau ihrer Berechnungsvorschrift aus den Berechnungsvorschriften für N und $\mathbb{N}$ angeht.

Für die Berechnungsterme werde ich wieder die Symbole t , N , $\mathbb{N}$ verwenden, die hier aber von den Funktionalen N und $\mathbb{N}$ bzw dem Term t zu unterscheiden sind.

BERECHNUNGSTERME

Berechnungsterme BT werden aufgebaut aus den Symbolen t , N , $\mathbb{N}$, $*$, Knoten (endlichen Folgen aus $\mathbb{N}$) und Klammern $(,)$ nach den Regeln:

- 1) t ist ein Berechnungsterm.
- 2) Ist s ein Berechnungsterm und a ein Knoten, so sind $N(s)$, $\mathbb{N}_a(s)$, s_a* ebenfalls Berechnungsterme.

Jedem Berechnungsterm s wird nun eine Funktion $\hat{s} : \mathbb{N} \rightarrow \mathbb{N}$ zugeordnet, so, daß $\hat{s}(l(c))$ gerade die Höhe des Berechnungsbaumes wird, die der entsprechenden Term hat.

Die Zuordnung ist definiert durch:

- 1) $\hat{t} : m \mapsto 0$
- 2) $\hat{N(s)} : m \mapsto (2^{m+1} - 1) + \hat{s}(m+1)$
- 3) $\hat{\mathbb{N}_a(s)} : m \mapsto 2^m(2^{l(a)} - 1) + \hat{s}(m + l(a))$
- 4) $\hat{s_a*} : m \mapsto \hat{s}(m + l(a))$

$\hat{s}$ ist für jeden Berechnungsterm s eine elementare, schwach monoton steigende Funktion von $\mathbb{N}$ nach $\mathbb{N}$. Bei geeigneter Codierung der Berechnungsterme ist sogar $h(\ulcorner s \urcorner, m) := \hat{s}(m)$ elementar.

Jedem im Berechnungsbaum von $N(t)$ auftretenden Term s kann nun eindeutig ein Berechnungsterm zugeordnet werden. Der dem Term s

entsprechende Berechnungsterm wird ebenfalls mit s bezeichnet. Dem Wert von s an der Stelle a , s_a , wird dann die Zahl $\widehat{s}(l(a))$ zugeordnet. Diese Zahlen werden beim Absteigen im Baum (Wurzel oben) echt kleiner. Denn:

Tabelle:

Terme im Baum/ Berechnungsterme	Nachfolger im Baum/ Berechnungsterme
1) $N(s) = f(s_{\Delta}, s_0)$	s_{Δ}, s_0
2) $M_i(s)_b = g(m(s, i, b))$	s_c mit $l(c) \leq l(b)+1$ $N(s_{j*})_c$ mit $l(c)+1 \leq l(b)$ $N(s_{0j*})_c$ mit $l(c)+2 \leq l(b)$
3) $N(s)_a = N(M_a(s))_{\Delta}$ $= f(M_a(s)_{\Delta}, M_a(s)_0)$	$M_a(s)_{\Delta}, M_a(s)_0$
4) $M_{a*i}(s)_b = M_i(M_a(s))_b$ $= g(m(M_a(s), i, b))$	$M_a(s)_c$ mit $l(c) \leq l(b)+1$ $N(M_a(s)_{j*})_c$ mit $l(c)+1 \leq l(b)$ $N(M_a(s)_{0j*})_c$ mit $l(c)+2 \leq l(b)$

Es gelten die Abschätzungen:

- $\widehat{N}(s)(0) = 1 + \widehat{s}(1) > \widehat{s}(1) \geq \widehat{s}(0)$
- $\widehat{M}_1(s)(l(b)) = 2^{l(b)} + \widehat{s}(l(b)+1)$ also:
 $\widehat{M}_i(s)(l(b)) > \widehat{s}(l(c))$ für $l(c) \leq l(b)+1$
 $\widehat{M}_i(s)(l(b)) > (2^{l(c)+1} - 1) + \widehat{s}(l(c)+2) = \widehat{N}(s_{i*})(l(c))$
für $l(c)+1 \leq l(b)$
 $\widehat{M}_i(s)(l(b)) > (2^{l(c)+1} - 1) + \widehat{s}(l(c)+3) = \widehat{N}(s_{0i*})(l(c))$
für $l(c)+2 \leq l(b)$

$$\begin{aligned}
 3) \quad \widehat{N(s)}(l(a)) &= 2^{l(a)+1} - 1 + \widehat{s}(l(a)+1) \text{ also:} \\
 \widehat{N(s)}(l(a)) &> 2^{l(s)+1} - 2 + \widehat{s}(l(a)+1) = \widehat{N_a(s)}(1) \\
 \widehat{N(s)}(l(a)) &> 2^{l(s)} - 2 + \widehat{s}(l(a)) = \widehat{N_a(s)}(0)
 \end{aligned}$$

$$\begin{aligned}
 4) \quad \widehat{N_{a * i}(s)}(l(b)) &= 2^{l(b)}(2^{l(a)+1} - 1) + \widehat{s}(l(b)+l(a)+1) \text{ also:} \\
 \widehat{N_{a * i}(s)}(l(b)) &> 2^{l(b)+1}(2^{l(a)} - 1) + \widehat{s}(l(a)+l(b)+1) \\
 &\geq 2^{l(c)}(2^{l(a)} - 1) + \widehat{s}(l(a)+l(b)) = \widehat{N_a(s)}(l(c)) \\
 &\quad \text{für } l(c) \leq l(b)+1
 \end{aligned}$$

$$\begin{aligned}
 \widehat{N_{a * i}(s)}(l(b)) &> 2^{l(b)} - 1 + 2^{l(b)+1}(2^{l(a)} - 1) + \widehat{s}(l(a)+l(b)+1) \\
 &\geq 2^{l(c)+1} - 1 + 2^{l(c)+2}(2^{l(a)} - 1) + \widehat{s}(l(a)+l(c)+2) \\
 &= \widehat{N(N_a(s)_i)}(l(c)) \quad \text{für } l(c)+1 \leq l(b)
 \end{aligned}$$

$$\begin{aligned}
 \widehat{N_{a * i}(s)}(l(b)) &> 2^{l(b)} - 1 + 2^{l(b)+1}(2^{l(a)} - 1) + \widehat{s}(l(a)+l(b)+1) \\
 &\geq 2^{l(c)+2} - 1 + 2^{l(c)+3}(2^{l(a)} - 1) + \widehat{s}(l(a)+l(c)+3) \\
 &= \widehat{N(N_a(s)_{oi})}(l(c)) \quad \text{für } l(c)+2 \leq l(b)
 \end{aligned}$$

Damit gilt:

SATZ 1

Das Funktional N ist wohldefiniert und stetig.

Bemerkung:

Die oben gegebene Schranke $\widehat{s}(l(a))$ für die Höhe des Berechnungsbaumes von s_a ist scharf. D.h. es gibt Terme s und Knoten a , für die $\widehat{s}(l(a))$ genau die Höhe des Berechnungsbaumes ist.

Beispiel: für $t = ((\dots((x_0 x_1) x_2) x_3 \dots) x_n)$ im λ -Kalkül, ist

bei der Berechnung von $N(t)_a$ mit $a = \langle 0, \dots, 0 \rangle$, $l(a) < n-1$ ist

$\widehat{N(t)}(l(s)) = 2^{l(a)+1} - 1$ die exakte Höhe des Berechnungsbaumes, also die Rekursionstiefe.

BERECHNUNGSKOMPLEXITÄT

Den Beweis, daß N und M elementare Funktionale sind, hier exakt zu führen, wäre extrem aufwendig und langweilig. Es sollen deshalb nur die wichtigsten Beweisschritte skizziert werden.

Man versteht die Funktionale N und M zuerst mit einem zusätzlichen Parameter k , der als Schranke auf die Rekursionstiefe benutzt wird, und setzt $N(k, s, a) = \emptyset$, falls diese Schranke überschritten wird.

Also etwa $N(0, s, a) := \emptyset$

$$N(k, s, a) := \begin{cases} f(s_{\Delta}, s_0) & \text{für } k > 0, a = \Delta \\ f(M(k-1, s, a), N(k-1, s, a, 0)) & \text{für } a \neq \Delta \end{cases}$$

 ... ect.

Die so definierten Funktionen sind dann durch beschränkte Rekursion definierbar und somit elementar.

Da die Funktion $h(s, b) := \widehat{s}(l(b))$ ebenfalls elementar ist kann dann N und M als Komposition elementarer Funktionen definiert werden.

REDUKTION UND REP

Die Unterschiede zu früheren Definitionen (Kap.I, Kap.III.1), die sich durch die Rep-Regel ergeben, werden alle so behandelt als handele es sich bei "Rep" um eine kritische Beseitigung, allerdings mit leerem Hauptteil (Wertigkeit (0)). Dadurch ist das erste (und einzige) Argument der Rep-Regel eine Nebenformel, entgegen unserer sonstigen Konvention.

Die Definition von Reduktionsniveau wird dadurch so modifiziert:
 Direktes Reduktionsniveau

$d(t) := n(t_0, *)$ falls t_{Δ} eine Beseitigung und t_0 eine Einführung, eine kritische Beseitigung oder Rep.

$d(t) := 0$ sonst.

Das Reduktionsniveau $r(t)$ ist die kleinste Zahl größer als die direkten Reduktionsniveaus aller Teilterme von t .

Weiter sei $\vdash$ (siehe Seite 13) modifiziert durch:

für alle t gilt $t \vdash \text{Rep}(t)$.

SATZ 2 $N(t)$ ist normal

Beweis:

Ist $N(t)$ nicht normal, so gibt es einen Knoten a so, daß $N(t)_{a*} \not\vdash s$ nach einer Regel 2) oder 3), für einen Term s .
Ist nun $a = \wedge$, so verifiziert man leicht anhand der Definition von $N(t)_{\wedge}$ und der $N_i(t)$ (die die Knoten in $N(t)$ über $\wedge$ bestimmen), daß dies nicht möglich ist.
Ist $a = a' * k$, so ist $N(t)_{a*} = N(N_{a'}(t))$ (nach Lemma I) und die Behauptung folgt wie oben.

FUNDIERTHEIT

Während die bisher bewiesenen Aussagen: Wohldefiniertheit, Stetigkeit, Normalität für beliebige Terme gelten, insbesondere auch für nicht getypte Terme oder nicht fundierte Terme, ist nicht zu erwarten, daß $N(t)$ fundiert ist, wenn nicht auch t fundiert und getypt ist.

Deshalb seien im folgenden die Terme wieder getypt und fundiert und es gelten die Normalisierungsbedingungen.

LEMMA 2

Sind s und t Terme mit $r(s) \leq k$ und $r(t) \leq k$, sowie $n(s) < k$, weiter x eine Variable vom selben Typ wie s , dann ist $r(t_x[s]) \leq k$.

Beweis:

Induktion über $\|t\|$, Fallunterscheidung nach $t_{\wedge}$:

- 1) $t_{\wedge}$ Variable $\Rightarrow$ trivial.
- 2) $t_{\wedge}$ Einführung $\Rightarrow r(t_x[s]) = \sup_i \{r((t_i)_{x[s]})\} \leq k$
nach Induktionsvor.
- 3) $t_{\wedge}$ Beseitigung $\Rightarrow r(t_x[s]) = \max\{d(t_x[s])+1, \sup\{r(t_i)_{x[s]}\}\}$
Ist $t_0 \neq x$, so gilt wegen $r(t) \leq k$, $d(t_x[s]) = d(t) < k$.
Ist $t_0 = x$, so ist $d(t_x[s]) \leq n(s) < k$.
Zusammen mit der Induktionsvor. folgt $r(t_x[s]) \leq k$.

KOROLLAR

Sei H, H' eine Kontraktionsvorschrift,

$$H = B(E(\dots, (\vec{x}_i)T_i(\vec{x}_i), \dots), \dots, (\vec{y}_j)S_j(\vec{y}_j), \dots) ,$$

t_i und s_j seien normale Terme, $\vec{x}_i$ und $\vec{y}_j$ Variablenlisten

$$\text{und } h' = H' \left[\left[(\vec{x}_i)t_i \right] \dots \left[(\vec{y}_j)s_j \right] \dots \right]$$

dann ist $r(h') \leq n(E(\dots, (\vec{x}_i)t_i, \dots))$.

Beweis:

Induktion nach dem Aufbau von H' :

1) $H' = T_i$ oder S_j trivial, da t_i und s_j normal.

2) $H' = T(H_1, H_2, \dots)$

$$\text{mit } h_k := H_k \left[\left[(\vec{x}_i)t_i \right] \dots \left[(\vec{y}_j)s_j \right] \dots \right]$$

gilt $r(h_k) \leq n(E(\dots))$ nach Induktionsvoraussetzung und

und $n(h_k) < n(E(\dots))$ nach der ersten Normalisierungsbed.

die Behauptung folgt dann mit obigem Lemma.

LEMMA 3

Sei t ein Term, t_Δ eine nicht kritische Beseitigung und

$r(t) \leq n(t_{0*})$, dann hat $N(t_{0*})$ die Form $\underbrace{\text{Rep}(\dots \text{Rep}(t')\dots)}_n$

mit $t'_\Delta =$ normale unkritische Beseitigung oder Variable, $n \in \mathbb{N}$.

Beweis:

Induktion nach $\|t_{0*}\|$

Fallunterscheidung nach t_{0*} :

1) t_{0*} Variable, normale unkritische Beseitigung $\Rightarrow$ Behauptung für $n=0$.

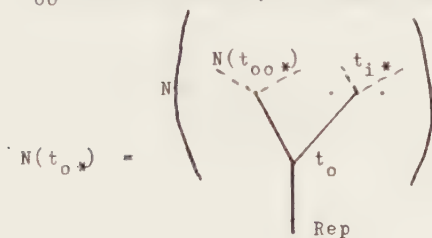
2) t_{0*} Einführung, kritische Beseitigung oder Rep $\Rightarrow r(t) > n(t_{0*})$.

3) t_{0*} nicht kritische Beseitigung: Fallunterscheidung nach t_{00} :

t_{00} Variable, normale Beseitigung, Einführung, kritische Beseitigung oder Rep $\Rightarrow$

Behauptung folgt wie bei 1) und 2) ($n(t_{0*}) < n(t_{00*})$).

t_{00} nicht normale unkritische Beseitigung $\Rightarrow$



Es ist $\|t_{0*}\| < \|t_{00*}\|$ und $r(t_{0*}) < r(t) \leq n(t_{0*}) \leq n(t_{00*})$ nach Normalisierungsbedingung 2. Aus der Induktionsvor. folgt

$N(t_{00*}) = \underbrace{\text{Rep}(\dots \text{Rep}(t')\dots)}_n$ mit $t' =$ unkritische normale

Beseitigung oder Variable. Nach Definition von N folgt sofort

$N(t_{0*}) = \text{Rep}(\underbrace{\text{Rep}(\dots \text{Rep}(t')\dots)}_n)$

SATZ 3

Ist t ein fundierter und getypter Term, so auch $N(t)$ und $t \triangleright N(t)$.

Beweis:

Daß $N(t)$ getypt ist und $t \triangleright N(t)$, ergibt sich aus der Definition von N leicht mit Induktion nach $\|t\|$. Dies wird hier nicht ausgeführt. Die Fundiertheit von $N(t)$ wird mit geschachtelter Induktion bewiesen:

Induktion nach $r(t)$

Induktion nach $\|t\|$

Induktion nach $\|t_{0*}\|$

Vorbemerkung: nach Lemma 1 ist $N(t)_{i*} = N(N_i(t))$. Nun folgt die Fundiertheit von $N(t)$ sofort, wenn alle $N(t)_{i*}$ fundiert sind. So wird in den meisten der folgenden Fälle die Behauptung durch Anwendung der Induktionsvoraussetzung auf $N_i(t)$ bewiesen.

Fallunterscheidung nach t_{Δ} :

$t_{\Delta} \neq$ nicht normale Beseitigung:

Mit Induktionsvor. da $\|N_i(t)\| = \|t_{j*}\| < \|t\|$

$t_{\Delta} =$ nicht normale Beseitigung:

Fallunterscheidung nach t_0 :

t'_0 = Variable oder unkritische normale Beseitigung $\Rightarrow$

$$N(t') = \begin{array}{c} N(N(t_{0*})) \quad \dots \quad N(t_{j*}) \\ \diagdown \quad \quad \diagup \\ t_{\Delta} \end{array} \dots$$

$N(t_{j*})$ ist fundiert nach IV., da $\|t_{j*}\| < \|t\|$
und $N(N(t_{0*})) = N(t_{0*})$ ist ebenso nach IV fundiert.
 $\Rightarrow N(t')$ fundiert.

t'_0 = Einführung $\Rightarrow$

mit Lemma 3 folgt dann sofort, daß $r(t) > n(t_{0*})$.

Wie im obigen Fall t_0 = Einführung folgt die Fundiertheit mit Induktion nach $r(t)$.

t'_0 = Rep oder kritische normale Beseitigung $\Rightarrow$

$$N(t') = \begin{array}{c} t'_{00*} \quad \dots \quad t'_{j*} \\ \diagdown \quad \quad \diagup \\ t_{\Delta} \end{array} \quad \text{b.z.w.} \quad = \begin{array}{c} t'_{00*} \quad \dots \quad \begin{array}{c} t'_{0i*} \quad t'_{j*} \\ \diagdown \quad \diagup \\ t_{\Delta} \end{array} \\ \diagdown \quad \quad \diagup \\ t'_0 \end{array} \dots$$

Rep

zu zeigen bleibt $N \left(\begin{array}{c} t'_{00*} \quad t'_{j*} \\ \diagdown \quad \diagup \\ t_{\Delta} \end{array} \right)$ b.z.w. $N \left(\begin{array}{c} t'_{0i*} \quad t'_{j*} \\ \diagdown \quad \diagup \\ t_{\Delta} \end{array} \right)$

ist fundiert.

Es liegt nahe dies induktiv zu beweisen. Man definiert hierzu die kritische Länge $kr(s)$ eines Termes s durch:

$$kr(s) := \begin{cases} \sup \{ kr(s_{i*}) \} + 1 & \text{falls } s_{\Delta} = \text{kritische Beseitigung} \\ kr(s_{0*}) + 1 & \text{falls } s_{\Delta} = \text{Rep} \\ 0 & \text{sonst} \end{cases}$$

und schließt mit Induktion nach $kr(t'_0)$:

$kr(t') = 0$: gerade die oben schon behandelten Fälle.

$kr(t') > 0$: Es folgt die Fundiertheit aus der Induktionsvor.

$$\begin{aligned} \text{da : } kr(t'_{00*}) &< kr(t'_{0*}) & \text{für } t'_0 &= \text{Rep} \\ kr(t'_{0i*}) &< kr(t'_{0*}) & \text{für } t'_0 &= \text{kritische Bes.} \end{aligned}$$

HÖHENABSCHÄTZUNG

Sucht man den vorgehenden Satz konstruktiv zu verschärfen (in Analogie zu den Höhenabschätzungen von Kap.III,1.), kommt man zu einer Behauptung der Form :

Gibt es einen Knoten a mit $N(t)_a \neq \emptyset$, so gibt es einen Knoten a' mit $t_a \neq \emptyset$ und $l(a') \geq \mathcal{Y}(l(a))$, wobei $\mathcal{Y}(n)$ mit n über alle Grenzen wächst.

Dies ist zum einen eine Höhenabschätzung ($l(a) \leq \mathcal{Y}^{-1}(l(a'))$) und zum anderen gibt es zu solch einer Abschätzung eine triviale Konstruktionsmethode, um aus einem Ast der Länge $l(a)$ in $N(t)$ einen Ast der Länge $\mathcal{Y}^{-1}(l(a))$ in t zu konstruieren.

Denn $N(t)_a$ hängt nur von endlich vielen Werten t_c ab. Der gesuchte Knoten a' ist somit unter diesen c 's auffindbar.

Eine Höhenabschätzung auf analogem Wege zu gewinnen wie in Kap III.1., scheitert an der unterschiedlichen Reduktionsstrategie (innermost Reduktion gegenüber maximal innermost Reduktion).

Damit wird die Ungleichung für $\mathcal{Y}(\|t\|, r(t))$

aus Kap III.1. $\mathcal{Y}(n+1, r+1) \geq \eta(\mathcal{Y}(n, r+1)+1)$

zu $\mathcal{Y}(n+1, r+1) \geq \mathcal{Y}(\eta(\mathcal{Y}(n, r+1)), r)$.

Dies ergibt dann statt einer $\mathcal{E}_4$ Funktion eine allgemein rekursive Funktion.

Der Versuch nur die Anzahl der Rep-Regeln abzuschätzen, um dann mit der Eindeutigkeit der Normalform zu argumentieren, scheitert daran, daß sich für die Rep-Regeln alleine, bedingt durch die Reduktionsstrategie, eine ähnliche Gleichung ergibt.

Eine durchführbare Alternative ist, die Termbäume direkt mit Typen anzureichern und die Funktionale N und $\bar{N}$ dahingehend zu modifizieren, daß nur noch Redexe mit direktem Reduktionsrang k reduziert werden. (Das Unterstreichen für "normaler Teilterm" muß ersetzt werden durch eine Zahl k für " Teilterm mit Rang $\leq k$)

Die so entstehenden Funktionale N^k und $\bar{N}^k$ reduzieren dann stetig einen Term so, daß $r(t) \leq k \Rightarrow r(N^k(t)) < k$.

Das Funktional N erhält man als $N^1(N^2(\dots N^k(t)\dots))$.

Für ein solches Funktional lassen sich die Höhenabschätzungen analog durchführen.

LITERATUR:

- [1] Aczel, P.: A General Church-Rosser Theorem.
Unveröffentlicht, 1978 .
- [2] Gandy, R.O.: Proofs of Strong Normalization.
In: To H.B.Curry: Essays on Combinatory Logic, Lambda Calculus
and Formalism. (eds. Seldin/Hindley), Academic Press 1980 .
- [3] Girard, J.Y.: Theorem de Church-Rosser pour un systeme de termes
infinis. Unveröffentlicht, 1973.
- [4] Kreisel, Mints, Simpson : The Use of Abstract Language in
elementary Metamathematics: Some pedagogic Examples.
In: Lecture Notes 453, Springer Verlag 1972 .
- [5] Maaß, W.: Eine Funktionalinterpretation der prädikativen Analysis.
Dissertation München 1974.
- [6] Maaß, W.: Church Rosser Theorem für λ -Kalküle mit unendlich
langen Termen. In: Lecture Notes 500, Springer Verlag 1974 .
- [7] Mints, G.E.: Finite Investigations of Transfinite Derivations.
In: Journal of Soviet Mathematics 10 , 1978 .
- [8] Schwichtenberg, H.: Complexity of Normalization in the pure
typed lambda-calculus. In: The L.E.J.Brouwer Centenary
Symposium. (eds. Troelstra/Van Dalen), North Holland,
Amsterdam, Studies in Logic 110, 1982 .
- [9] Tait, W.W.: Infinitely Long Terms of Transfinite Type.
In: Formal Systems and Recursive Functions. (eds. Crossley/
Dummet), North Holland, Amsterdam 1965 .

Lebenslauf:

Ich wurde am 28.5.57 in Dornau geboren.

Ab 1963 besuchte ich die Volksschule in Mespelbrunn/Spessart, später in Stockstadt/Main . 1970 Übertritt ins Karl-Theodor von Dalberg Gymnasium in Aschaffenburg, wo ich 1976 das Abitur ablegte.

Beginn des Studiums der Mathematik/Physik an der Universität Würzburg im Wintersemester '76/'77 . Nach dem Vordiplom einjähriges Auslandsstudium an der ETH-Zürich. Zum Wintersemester '79 Wechsel an die TU-München. Ab Sommersemester '80 Studium an der LMU-München. Nach dem Studienabschluß im September '81 an der LMU-München mit dem Diplom in Mathematik wurde ich zum Zivildienst einberufen. Während des Zivildienstes Nebentätigkeit in der Software-Entwicklung bei Fa. Letterphot.

Seit 1983 Arbeit an der Dissertation .

